High Performance Computing

# Assignment 1 – Brianna Drew

# High Level basic theory

1. ***Partially driven by the approaching end of Moore’s law, and partially driven by changes in semiconductor manufacturing, several engineering research papers have attempted to propose alternatives to the semiconductor node sizes measured in nanometres. (E.g. Ryzen 3 are made on a 7nm node, Intel is still using 10 nm). Discuss why a new measurement system is necessary, and what you think are appropriate alternatives and why, either discuss your own idea or use one of the existing papers as a basis for your opinion. (TL;DR so you know you’re on the right track: it used to be that since transistors made in 2D planes, and the height and width shrunk at the same rate, one number squared corresponded to the inverse of the density, so 7nm was 2x the transistor density of 10nm, 5nm is 2x the density of 7nm, but that underlined bit is not the case anymore and so we need a better measure). Up to 1000 words answer.***

Moore’s Law refers to a doubling of transistors on a chip with each process generation. This law has named each successive process node approximately 0.7 times smaller than the previous one – a linear scaling that implies a doubling of density. There was 90 nm, 65 nm, 45 nm, 32 nm – each enabling the packing of twice the number of transistors in a given area than was possible with the previous node. Because of the increasing difficulty of further scaling – some companies have abandoned this rule, yet have continued to advance node names, even in cases where there was minimal or no density increase. The result is that node names have become a poor indicator of where a process stands on the Moore’s Law curve. The industry needs a standardized density metric. The challenge is in the increasing complexity of semiconductor processes, and in the variety of designs. What is really needed is an absolute measure of transistors in a given area (per mm2). A metric used in the past but discontinued a while back is based on the transistor density of standard logic cells and includes weighing factors that account for typical designs. A very simple example is a 2-input NAND cell (4 transistors) – and one that is more complex but also very common: a scan flip flop (SFF). This leads to a previously accepted formula for transistor density:

* 1. x NAND2 Tr Count + 0.4 x Scan Flip Flop Tr Count = # Transistors/mm2

NAND2 Cell Area Scan Flip Flop Cell Area

*(The weightings 0.6 and 0.4 reflect the ratio of very small and very large cells in typical designs.)*

Every chip maker, when referring to a process node, should disclose its logic transistor density in units of MTr/mm2 (millions of transistors per square millimetre) as measured by this simple formula. Given the wide variety of SRAM-to-logic ratios in different chips, it is best to report SRAM cell size separately, next to the NAND+SFF density metric.

1. ***Based on*** [***https://www.sharcnet.ca/my/systems/show/114***](https://www.sharcnet.ca/my/systems/show/114) ***Graham (the main Sharcnet cluster) uses the following for compute nodes:***

* ***Intel E5-2683 v4 (Broadwell) @ 2.1 GHz X2 (so dual socket 16 core CPUs)***
* ***2 × NVIDIA Pascal P100 GPUs (12GB HBM2) (only some nodes)***
* ***Memory: 128.0 GB (RAM)***
* ***Local storage: 1.2TB.***

***These are getting old. A reasonable replacement for this would be something using a dual socket AMD Epyc 64 core system (any of the 7763 or 7713 for our purposes they’re similar, but for this question assume a 7763), with 2 NVIDIA A100 Accelerators, and 256 GB of memory, using PCIe 4.0 SSDs around 2 TB.***

***Sharcnet is configured with 800 nodes using no GPUs, and about 150 with GPUs (and about 100 other machines which we will ignore for now, that are for testing, login, extra memory configurations etc).***

***Estimate the cost of replacing the 950 nodes (800 nodes without a GPU 150 with a GPU) mentioned with their modern equivalents and attempt to find (from vendors) something close to the mentioned configuration to justify your answer (Fujistu, IBM, NUDT, Dell EMC, Cray/HPE and Huawei are the biggest vendors in this business. Note: don’t call companies and ask about this, we’re not trying to bother them.***



1. ***Estimate the power cost for running one of the above systems (Dual AMD Epyc 64 core + 2 NVIDIA A100s). Don’t worry about trying to price out interconnects to other servers or the cost of a rack itself.***

***Assuming the power supplies in your system draw full power 100% of the time (they wouldn’t in the real world) what is the power cost per week of using electricity rates from:***

[***https://www.torontohydro.com/for-business/rates***](https://www.torontohydro.com/for-business/rates) ***(you need to do some simple math to figure out the cost for each of the 168 hours in a week and multiply by electricity costs in kilo Watt hours into the same form as power in Watts).***

Dual AMD Epyc 64-core - Default TPD 225 Watts

NVidia A100 - 400 Watts

Total kwh = (225 \* 2 CPU) + (2 \* 400) = 1250 watts/hour

1250 watts per hour \* 7 days \* 24 hr/day = 210 kwh/week

Tier 1 Non-residential customers up to 750 kWh/month is 9.8 cents/kWh

Graphical user interface, text, application, email

Description automatically generated

So, 210 kWh \* .098 $/kWh = **$20.58**

# Programming

1. ***Numerical method in C++ on Linux/CentOS (If you are using Python and WSL do the same thing, but well, in python).***

***We’re going to do our own cpu benchmarking program.  
  
This is a Linear Algebra – (solutions to PDE’s), type problem.***

***Write a C++ program that generates random 10x10 systems of linear equations and solve with with Gauss Jordan elimination. (Note this means that your actual matrix will have 11 columns and 10 rows).***

[***http://www.cplusplus.com/reference/cstdlib/rand/***](http://www.cplusplus.com/reference/cstdlib/rand/)

[***https://www.geeksforgeeks.org/program-for-gauss-jordan-elimination-method/***](https://www.geeksforgeeks.org/program-for-gauss-jordan-elimination-method/) ***(you probably need it you can find ‘how to generate random matrices in C++ on the web too, though fair warning modern C++ isn’t supported in the built in GCC compiler on CentOS so some random number generators that are new and fast don’t work)***

1. ***Throughput – every 1 second print off how many times it was able to run the solutions, run this for 10 seconds.***
2. ***Capacity computing (job time) – calculate how long it takes for 100, 200, 500, 1000, 2500, 5000, 7500 and 10 000 runs of the calculation. (If that takes absurdly long on your machine pick smaller numbers, I don’t actually care what it is, it seems like 10 000 should take about 5 minutes). Plot the result (feel free to use excel for the plot).***

A picture containing chart

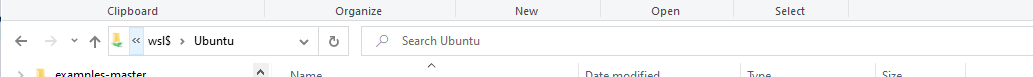
Description automatically generated

# Linux Commands

1. ***Demonstrate a basic understanding of Linux Commands based on the following***

[***https://www.thegeekstuff.com/2010/11/50-linux-commands/?utm\_source=feedburner***](https://www.thegeekstuff.com/2010/11/50-linux-commands/?utm_source=feedburner)

***(If you’re using WSL and you need to find the files it outputs, make sure you start WSL and then in this box***

******

***On the left type*** [***\\wsl$***](file:///\\wsl$) ***which will then show you “Ubuntu” as a network location (this breaks if you’ve got VPN turned on btw, within that, your stuff is likely stored in /home/yourusername)***

1. ***Using the Linux pipe command, output the result of your program in 4 to a file (even if your program does not work you should know how to do this). Paste that file here.***

\*Files are included in the tar.gz file as they are too long to paste here (assn1\_1.cpp is Question 4 part a, assn1\_2.cpp is Question 4 part b).

1. ***While your program in 4 is running doing its thing (or something else is running, doesn’t matter what) use ncdu and top to see how much memory and what percentage of CPU your program uses. Show your output here. (You can do that with WSL if you need to).***

ncdu 1.16 ~ Use the arrow keys to navigate, press ? for help

--- /home/bdrew/COIS4350H/Assignment1 ---------------------------------------------

116.0 MiB [###########] q4\_a.txt

68.0 KiB [ ] q4\_b.txt

28.0 KiB [ ] q4\_b

28.0 KiB [ ] q4\_a

8.0 KiB [ ] grep.txt

4.0 KiB [ ] assn1\_1.cpp

4.0 KiB [ ] assn1\_2.cpp

Total disk usage: 116.1 MiB Apparent size: 116.1 MiB Items: 7

[bdrew@localhost Assignment1]$ top

top - 17:11:07 up 34 min, 1 user, load average: 0.36, 0.23, 0.19

Tasks: 228 total, 1 running, 227 sleeping, 0 stopped, 0 zombie

%Cpu(s): 0.8 us, 0.3 sy, 0.0 ni, 98.9 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st

MiB Mem : 808.7 total, 58.5 free, 364.3 used, 385.9 buff/cache

MiB Swap: 1536.0 total, 733.8 free, 802.2 used. 317.0 avail Mem

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND

2188 bdrew 20 0 3982580 138232 69176 S 5.3 16.7 1:23.02 gnome-s+

5396 bdrew 20 0 536972 46684 31752 S 0.7 5.6 0:26.19 gnome-t+

2384 bdrew 20 0 475024 1716 1312 S 0.3 0.2 0:00.17 gsd-sma+

1 root 20 0 252692 6220 4004 S 0.0 0.8 0:01.46 systemd

2 root 20 0 0 0 0 S 0.0 0.0 0:00.00 kthreadd

3 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu\_gp

4 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu\_par+

6 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker+

9 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 mm\_perc+

10 root 20 0 0 0 0 S 0.0 0.0 0:00.01 ksoftir+

11 root 20 0 0 0 0 I 0.0 0.0 0:00.54 rcu\_sch+

12 root rt 0 0 0 0 S 0.0 0.0 0:00.00 migrati+

13 root rt 0 0 0 0 S 0.0 0.0 0:00.00 watchdo+

14 root 20 0 0 0 0 S 0.0 0.0 0:00.00 cpuhp/0

15 root 20 0 0 0 0 S 0.0 0.0 0:00.00 cpuhp/1

16 root rt 0 0 0 0 S 0.0 0.0 0:00.00 watchdo+

17 root rt 0 0 0 0 S 0.0 0.0 0:00.00 migrati+

1. ***Use Grep to find all instances of the word “and” on the syllabus (you’ll need to save the syllabus to .txt first), and print out the lines they are on.***

16:The course will consist of weekly meetings Fridays 14:00-15:50 on ENW 111 and a lab Mondays from 14:00-14:50 in OC202. Check the

17:Academic timetable for updates to rooms and times.

28:power, memory access, or storage. Core topics include advanced computer architectures, programming for shared and distributed memory

29:machines, networking issues, caching, performance evaluation and parallel algorithms. Topics are supplemented with case studies.

32:Students will be expected to demonstrate mastery of high-performance concepts including both theory and programming practice.

33:Students will also be expected to research an area of HPC not covered in class and be able independently develop working code and solve a

37:High Performance Computing, Modern Systems and Practices 1st Edition, Authors: Thomas Sterling Matthew Anderson Maciej Brodowicz

42:and https://www.youtube.com/channel/UCCRmb5\_GMWT2hSlALHlwIMg

46:Assessments, Assignments and Tests:

47:Weekly Tech: Each week students will be expected to individually follow tutorials and show that they can test and minimally use each of the

49:Assignments: Assignments will be a combination of programming questions and theory questions to demonstrate understanding of the

50:material. Assignments cover Linux setup and use, parallel computing concepts, including algorithms and data structures, and cluster topics

51:including architecture and execution. Because much of the purpose of HPC is in solving numerical problems, students will be expected to learn

62:class and assignments

65:Weekly Labs: 10% total, best 10/11 labs. The last week will only have time for review questions and won't be a lab.

105:While there is no attendance mark in this course, students are expected to regularly attend and perform the laboratory exercises.

108:Students are expected and encouraged to bring a laptop to the lab sessions and to collaborate with other students. Assignments are to be

111:graduate students, without exception. The scope and nature of the projects is not the same between the two courses.

115:The Trent e-mail account is considered the official e-mail account and will be the only e-mail account used to communicate with students for

116:academic and administrative purposes. Students are responsible for ensuring that they monitor and maintain their Trent e-mail account and to

117:ensure that e-mail is accessed, read, and acted upon in a timely fashion. Students should be aware that e-mails from non-Trent accounts will

129:Academic dishonesty, which includes plagiarism and cheating, is an extremely serious academic offence and carries penalties varying from

130:failure on an assignment to expulsion from the University. Definitions, penalties, and procedures for dealing with plagiarism and cheating are

135:It is Trent University's intent to create an inclusive learning environment. If a student has a disability and documentation from a regulated health

136:care practitioner and feels that they may need accommodations to succeed in a course, the student should contact the Student Accessibility

139:Sharing and Distribution of Course Content

140:Students in this class should be aware that classroom activities (lecture, seminars, labs, etc.) may be recorded for teaching and learning

142:distributes course content in any way that breaches copyright legislation, privacy legislation, and/or this policy, the student will be subject to

143:disciplinary actions under the Student Charter of Rights and Responsibilities or the relevant Academic Integrity Policy, at a minimum, and may

146:here: https://www.trentu.ca/artsci/sites/trentu.ca.artsci/files/documents/Policy%20on%20Sharing%20and%20Distribution%20of%20Course%20Co

149:Student Absenteeism, Missed Tests and Examinations

150:Students are responsible for completing all course requirements, including attending classes and meeting assignment deadlines as specified on

152:Adjustments and deferrals to dates for participation, assignment submissions, tests, midterms and final examinations are not automatic. It is the

155:Students unable to participate (i.e., by video and/or audio) should email their instructors to request alternative arrangements for participation in

157:Students are required to be available for all tests, midterms and exams that are listed in their course syllabus and scheduled by their instructor or

159:and tests. Normally a doctor’s note or supporting documentation is not required; however, when a student’s success in the course or program

162:the student to make these arrangements in advance as per SAS guidelines, and to discuss accommodations of due dates with their instructors.

1. ***Tar up your assignment and submit it (name it for your trent username.tar.gz), on future assignments feel free to use zip or rar like a normal person.***