COMPUTER SCIENCE

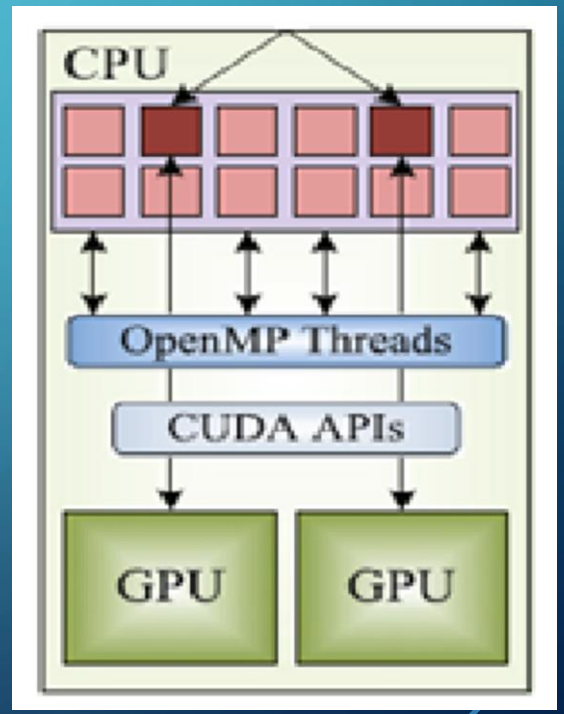
LESSON 3: THURSDAY SEPTEMBER 14TH , 2017

COMPUTER HARDWARE

- Transistors : semiconductors that amplify or switch electronic signals and power. Replaced Vacuum tubes, cut back on space, extra heat generation, and unreliability.
- Micro-processors : an integrated circuit with all the functions of a CPU. Accepts binary input data. Greatly reduced costs and improved efficiency. These are what smart phones and other hand-held devices use!
- Cores and threads: Every computer has cores and threads. A core is a component that DOES a task. Cores can be considered processing units. Each core then can be subdivided into threads, which do individual operations (like adding and subtracting)

BEYOND THE BASICS

- Parallel programming
  - NOT reserved for special machines – you can parallelize any code on almost any machine!
  - Instead of only using 1 core and 1 thread at a time, optimize the program by using multiple cores at the same time.
- Supercomputers and clusters, HPC
- GPU programming
  - A subcategory of parallel programming, often used for extreme visualizations and video games in particular



SOCIETAL AND ENVIRONMENTAL IMPACTS

WHO WE ARE AS A COMMUNITY

A MANY FACETED ISSUE

- Why do computers impact society? How are they different from other inventions and tools?

A MANY FACETED ISSUE

- Why do computers impact society? How are they different from other inventions and tools?

- **Ubiquity**
  - Computers are everywhere, and we interact on a passive level
- **Magnification**
  - 1 error can affect millions of people, explosion of information
- **Accessibility**
  - Information and communication
- **Reproducibility and Distributability**
- **Lack of Accountability**
  - Hiding behind a screen, or no one wants to take responsibility for a mistake
- **Temporality**
  - Both ways, everything needs to be done faster, AND information lasts forever
- **Spatiality**
  - Long distance learning, sharing data
- **Surveillability**
  - Not just camera and mics, but transactions
- **Changing Protocols in Communication**
  - Emails eliminating verbal and visual cues
- **Illusian of Precision/ Black box**
  - Assuming computers work to infinite accuracy, no checking required

GOOD OR BAD?

- Are the advent and implementation of computers and computer science globally good, bad, or neutral? Why?

LET'S TAKE AN EXAMPLE: INFORMATION

- Even just Wikipedia. What are some benefits and drawbacks?

LET'S TAKE AN EXAMPLE: INFORMATION

- Even just Wikipedia. What are some benefits and drawbacks?
  - Covers a wide variety of issues
  - Worldwide group of contributors and editors, some being experts
  - "Wisdom of Crowds"
  - Fast editing and updating

  - Uneven coverage, maybe emphasis on pop culture
  - Inaccuracy and poor writng
  - Hidden agendas
  - Vandalism

CHANGES IN THINKING

- Is technology used as a crutch in some ways?
- Are some skill sets less useful now?
- Has ways of thinking been made easier?

EASIER AND … LESS EMPLOYABLE?

- Take students like yourselves
- Students should be tech-literate, and so technology and computers should be incorporated into the curriculum to help you gain the skills you need to get the jobs you want.

EASIER AND … LESS EMPLOYABLE?

- Take students like yourselves
- Students should be tech-literate, and so technology and computers should be incorporated into the curriculum to help you gain the skills you need to get the jobs you want.
  - So… is there a standard on what equipment and what knowledge? An exit exam? What should and should not be covered?
  - It could lead to extra work for a teacher to incorporate this on their own
  - Unequal access for students, plagiarism, student isolation

THOSE WHO ARE OPPOSED: NEO-LUDDITE

- Neo-Luddites are those who are opposed to computerization, the Internet, and modern technology
- Why would some people be so opposed?

THOSE WHO ARE OPPOSED: NEO-LUDDITE

- Neo-Luddites are those who are opposed to computerization, the Internet, and modern technology
- Why would some people be so opposed?
  - Loss of employment
  - Manufactures a need (feeds capitalism)
  - Hurts the environment
  - Social Isolation
  - Strengthen government, big business, and capitalism/materialism
  - Dehumanizes (emphasis on buy, produce, sell)
  - Widens social inequality ("widen the gap" vs "a rising tide lifts all boats")
- What are your thoughts? What would an appropriate way of life be?

ENVIRONMENTAL IMPACT

- There's the obvious Silicon Valley and computer waste argument, but how else do computers and technology effect the environment?

ENVIRONMENTAL IMPACT

- There's the obvious Silicon Valley and computer waste argument, but how else do computers and technology effect the environment?
  - Actually leads to an INCREASE in paper usage!
  - Disassociate from reality: usually don't care about what we don't see/know
  - Being able to use technology and all it brings has many side affects and branches that use power, burn fossil fuels, and create waste: A/C and heat, transport of food, transport of waste
- What can help?

THE RISE OF PYTHON

WHAT IT'S BASED ON AND WHAT IT'S USED FOR

(I PROMISE IT'S NOT LIKE THE MOVIE "ANACONDA")

WHERE DID PYTHON COME FROM?

- We've talked about Fortran (1950s) and C/C++ (1970s), so when and were did Python come from and how is it different?
- Python was created in the 1980s by Guido van Rossum, who was working on building a language called ABC (general purpose language, created in the Netherlands, Amsterdam). Working on ABC inspired Guido van Rossum to create a language with all the benefits of ABC and none of the problems (he was literally inspired by frustration). His motivation for some of the choices in the new scripting language was to appeal to Unix and C users.

WHY PYTHON?

- The name “Python” descends from Guido van Rossum’s love of Monty Python’s Flying Circus (a British comedy group)

- The first version of Python was available in 1991, already an object-oriented system with modular capabilities.
- Python 2.0 was introduced in 2000, with some fancy visualization tools
- Python 3.0 was released in 2008, with one very important change: it wasn't backwards compatible, and many of the functions in 2.x were lost.
  - "There should be one –and preferably only one – obvious way to do it"
- 3.x are getting closer to the "Zen of Python": *Beautiful is better than ugly. Explicit is better than implicit. Simple is better than complex. Complex is better than complicated. Flat is better than nested. Sparse is better than dense. Readability counts. Special cases aren't special enough to break the rules. Although practicality beats purity. Errors should never pass silently. Unless explicitly silenced. In the face of ambiguity, refuse the temptation to guess. There should be one -- and preferably only one -- obvious way to do it. Although that way may not be obvious at first unless you're Dutch. Now is better than never. Although never is often better than *right* now. If the implementation is hard to explain, it's a bad idea. If the implementation is easy to explain, it may be a good idea. Namespaces are one honking great idea -- let's do more of those!*

ASSIGNMENT 1

- Report based
- Pick 1 major achievement, breakthrough, person, or pivotal moment in computer science history. Write a brief report (3-5 pages = 2000-3750 words) on the circumstances/background of the person or event and why it/they are/were important to the field. Remember to include the proper format for a report and have reputable sources!
- Due by Sunday Sept 17th 11:59pm via email submission to woodford@cita.utoronto.ca.

REFERENCES

- https://www.cs.usfca.edu/~brooks/S08classes/cs480/slides/evaluating.pdf
- http://cs.millersville.edu/~bliffick/compchar.html
- https://web.utk.edu/~nolt/radio/computer.htm (read at your own risk, there's no holds barred in this lecture note.)
  - A bit biased…the last paragraph is a bit much.
- https://www.autodesk.com/products/eagle/blog/e-waste-conundrum-can-biodegradable-printed-circuit-boards-help/
- https://www.python-course.eu/python3_history_and_philosophy.php