# COMPUTER SCIENCE LESSON 24+25, TUESDAY DEC 5<sup>TH</sup> + THURSDAY DEC 7<sup>TH</sup>

WWW.CITA.UTORONTO.CA/~WOODFORD

WOODFORD@CITA.UTORONTO.CA

 $\Box$ 

0

 $\bigcirc$ 

 $\cap$ 

### THIS WEEK IN CS AND STEM

- Preventing Quantum cracking: making quantum computing more secure and (this is embarrassing) as fast as current internet speeds.
  - <u>https://futurism.com/future-cybersecurity-quantum-encryption/</u>
- Al surpassing human capabilities: learning new languages without any references?
  - <u>https://futurism.com/researchers-created-ai-system-teaches-itself-new-languages/</u>
- HIV vaccine ready for human testing!!
  - <u>https://futurism.com/vaccine-hiv-tested-thousands-people/</u>

# ACSL COMPETITION

• Contest 1 needs to be completed by Dec 22

- Need to form teams (3 or 5?)
- Need to complete and send before Dec 22

### DICTIONARIES

- This is a whole new kind of data structure called a set. It looks like a list, but there are some major differences.
- Dictionaries can be indexed by ANYTHING not just integers. These variables are called keys. It is also defined using curly braces, not brackets or parentheses.
  - Try it out in the jupyter notebook "Lecture22\_23\_Sets\_Dictionaries.ipynb"

# INPUT AND OUTPUT FILES

- A lot of the data we've been using thus far has been input from the user on the command line or hard-coded into the file. This isn't reasonable for large data sets or complicated entry methods.
- It's typically easier for users to create an input file that contains all the data needed by a function. What are some things you would need to consider as the programmer for taking in data files?

### GETTING DATA FROM FILES (INPUT)

- Need to define the file to look for, and what to do with it. Then you can actually start using the information inside and parsing it.
  - Typically it would look like this:
  - F = open("filename.txt","r") #so now F is the open, readable file.
  - How do we read lines? How do we pull out data? Let's look at Lecture24\_25\_input.txt start up one of Spyder or Jupyter (at least one person per group should have spyder and one person should have jupyter – we're contrasting how they work)

### PRINTING DATA TO FILES

- Output works in a similar way. Now you can make your own file (or overwrite an existing one). Just make sure you specify that you're writing to the file!
  Ex. F = open("filename.txt", "w")
  F.write("Yo")
  F.close()
  - Always remember to close your files when you open them this way! It will save energy on your compute cores and help python kernels run a little faster – plus the variable you use for your file will become available again!

# SPLIT

 Input files are always read as strings. We know that we can use int() or float() to covert file types, but what if you don't know when descriptive text ends and numbers or input data begins?

Can use split(), as in:
 For line in f:
 print(line.split())

• What does this look like for some of the files we've used/made so far?

#### ARGPARSE

- What if you need to know where your input files are? Or maybe you want to let the user name their output files? How would we do this?
- Firstly, we'll need to start moving beyond running python in the ipython command line and start running it in the terminal. This will open a lot more usability for us (and also prepare us for Unit E).

#### ARGPARSE

• A bit more formalism for this one. Take a look at the reference material for this on the python docs page for full details.

Import argparse

• Lets take a look in Lecture 24\_25\_argparse.py

#### **ASSIGNMENT 9**

Code/report

Follow the instructions in Assignment9.txt. It contains your pre and post condition – remember that it's up to you to define the "how". I suggest using .py files (ie scripting) over notebooks. You should submit a working script, a unit testing script if you modularized your code, and a short report explaining your logic (no more than 1 pg, at least 1 paragraph).

• Due Sunday Dec 10<sup>th</sup> by 11:59pm via email to woodford@cita.utoronto.ca

# REFERENCES

- <u>https://docs.python.org/3/library/filesys.html</u>
- <u>https://docs.python.org/3/library/functions.html#open</u>
- https://docs.python.org/3/iibrary/argparse.nimi
- <u>https://docs.python.org/3/tutorial/inputoutput.html</u>

# UPCOMING LECTURES

• ROM Speaks Dec 4<sup>th</sup>, 5<sup>th</sup> : Canada's Oceans, Towards 2020

- <u>https://www.rom.on.ca/en/whats-on/canadas-oceans-towards-</u>
  <u>2020?utm\_medium=email&utm\_source=engagingnetworks&utm\_campaign=CanadaOce</u>
  <u>ans&utm\_content=Canada+Oceans</u>
- AstroTours Dec 7<sup>th</sup> : Gravitational Waves, Sirens of the Universe
  - <u>http://www.astro.utoronto.ca/astrotours/?page\_id=392</u>