

CSCA48

Introduction to Computer Science II

Summer 2022

About your Instructor

- Purva R. Gawde
 - First name pronounced as Poorva
- Originally from....
 - India
- Interested in
 - Computer Science, Biology, Psychology, Philosophy, Math, Stats
- Hobbies
 - Sketching
 - Reading
 - Cooking
 - Sleeping
- Masters and PhD from Kent State university
- Research project with Summa Healthcare, Ohio
- Lecturer and course developer at Kent State University
- Postdoc researcher and course developer with Ryerson
- With UTSC since 2020..

Why I love Computer Science

- I love art!! (What does that have to do with computers?!)
 - Fuels creativity
 - I can create something abstract unpredictable art with computer!!
 - Example: AARON, Deep Dream
- Challenges me!
 - An ER doctor inspired me to think about following problem:
 - Automated system to detect heart diseases
 - Without invasive procedures!!
 - PhD dissertation..
- Computer Science is everywhere and it Inspires me to keep learning :)



Moonage Daydream: art created by Deep Dream. Photograph: Deep Dream

Questions??

- We love questions... so, let's start there
- Raise your hand if you're excited about learning computer science...
- Now put your hand down if you're excited about this course...
- Let's cover some common questions:
 - What is this course about?
 - What will we learn?
 - Why is it important?
 - What skills will you develop? What challenges will you face?
 - What do you have to do to succeed here?

What is this course about?

This course is about
Sculpting!

You learned that in A08!!

This course is NOT ABOUT PROGRAMMING

Though we have to do some of that



This course is about..

- Let's relate sculpting with CS..
- Sculpting is more closer to (?):
 - Implementing an algorithm
 - programming
- Cooking from recipes! Or Folding Origami! Are other examples of carrying out an algorithm



If you are asked to sculpt an elephant, how do you



The really interesting part about sculpting



*In this course you will learn a set
of fundamental tools and
techniques for solving problems
using computers*

You will need what you learn here
for any CS-related task!



The problem solving skill required to sculpt a piece

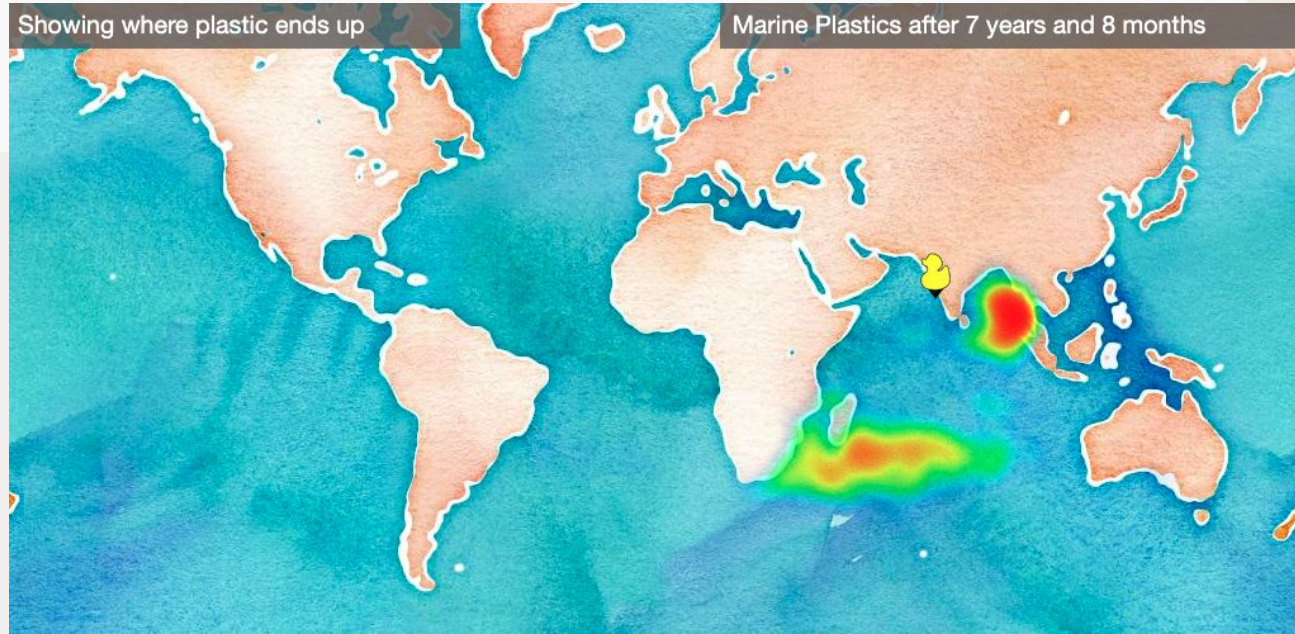
What kind of problems we solve with computer science?

- **Biology** - How to model genetics and help diagnose and cure disease
- **Architecture** - How to design better, lighter, stronger homes, roads, bridges
- **Movies** - How to create life-like simulated animals, environments, worlds
- **Transportation** - How to reduce gridlock, how to improve safety
- **Education** - How to improve education and make it accessible
- **Environment** - How to model our planet so we can understand our effect on it
- **Music** - How to record a music album when the band is spread across the world
- **Cooking** - How to come up with good, new combinations of ingredients
- **Data analytics** - Stats + computer science + machine learning!
- **Business Intelligence** - Market analysis and prediction, optimizing investments

Environment

Cleaning up the 5.25 billion pieces of plastic polluting our oceans

Plastic Adrift by Erik van Sebille



Research Citation

Van Sebille E, England MH, Froyland G. Origin, dynamics and evolution of ocean garbage patches from observed surface drifters. Environmental Research Letters. 2012 Dec 19;7(4):044040.

If not programming then what to expect?

- Each topic we will cover is **one fundamental component** of how we solve pretty much any kind of problem in Computer Science
- We will **link each topic covered with the previous one** so you understand completely how these ideas and concepts fit together
- We will **link every topic we cover to courses you will take after A48**, so you know where it will be used, and where you will learn more about it
- We will talk about **current applications and research** illustrating each technique we're learning about
- We will discuss **social, ethical, and professional issues** you will encounter in your career, so you become a solid, ethical, professional individual
- We (YOU & we) will build the right atmosphere for you to carry through your career – **a welcoming, supportive, and fun place for you to learn and grow**

What *you need to do* to make the most of this course

- YOU will have to **work hard**: Attend lectures, practicals, and do your exercises!
- The **programming language is C** – a language that was designed to let YOU understand exactly what your code is doing at every step
- We will cover all the aspects of C you need to understand, but **YOU have to practice** and develop the ability to use it well
- We will offer all the help we can – **plenty of support** in office hours, a forum, and plenty of exercises for you to try and practice. YOU have to use them.
- Your hard work will pay up – YOU will develop a much stronger **ability to solve problems**, and implement a solution you fully understand
- Your hard work will **make your life easier** next year in **B07, B63, B36, and B09** (you can thank yourself later!)
- Your programming knowledge after 1st year will be on par with CS students from Waterloo, Stanford, Berkeley...

TIOBE Index

The Importance of Being Earnest

The TIOBE programming community index is a measure of [popularity of programming languages](#), created and maintained by TIOBE Software BV, based in Eindhoven, the Netherlands









TIOBE Index for April 2022

April Headline: MATLAB about to drop out of the top 20

Good old MATLAB is about to drop out of the top 20 for the first time in more than 10 years. The MATLAB programming language is mainly used in the numerical analysis domain. It is often combined with Simulink models, which are from the same MathWorks company. Although MATLAB has a biannual release cycle, the language doesn't evolve that much. And since MATLAB licenses are rather expensive, alternatives are catching up quickly now. Its main competitors are Python (currently number 1) and Julia (moving from position 32 to position 26 this month). --Paul Jansen CEO TIOBE Software

The TIOBE Programming Community index is an indicator of the popularity of programming languages. The index is updated once a month. The ratings are based on the number of skilled engineers world-wide, courses and third party vendors. Popular search engines such as Google, Bing, Yahoo!, Wikipedia, Amazon, YouTube and Baidu are used to calculate the ratings. It is important to note that the TIOBE index is not about the *best* programming language or the language in which *most lines of code* have been written.

The index can be used to check whether your programming skills are still up to date or to make a strategic decision about what programming language should be adopted when starting to build a new software system. The definition of the TIOBE index can be found [here](#).

Apr 2022	Apr 2021	Change	Programming Language		Ratings	Change
1	3	▲	 Python		13.92%	+2.88%
2	1	▼	 C		12.71%	-1.61%
3	2	▼	 Java		10.82%	-0.41%
4	4		 C++		8.28%	+1.14%
5	5		 C#		6.82%	+1.91%
6	6		 Visual Basic		5.40%	+0.85%
7	7		 JavaScript		2.41%	-0.03%
8	8		 Assembly language		2.35%	+0.03%

What *we assume* you have learned thus far

- How to go about **taking an algorithm description and turning it into a program** **Variables**, what they are, how to use them
- How to use **conditional statements** to control what your program does
- How to use **loops** to carry out repeated operations
- How **functions** work: input arguments, return values, and how to use functions to get work done
- How to **break a complex algorithm into functions that work together** to solve the given task
- How to **debug and test a program** to make sure it is correct and works as intended

What *we expect* from you

- That you will *make the most of every opportunity we provide* for you to learn
- That you will *behave professionally and respectfully* toward everyone else in the course: We are all colleagues here
- That you will *do your own work, to the best of your ability*, and *not claim anyone else's work as your own*.
- That you will, to the degree that you can, *help your colleagues understand any ideas and concepts you have understood yourself*
- That you will use the forum for what it's meant: *Asking for help* understanding the course material, the problems we give out, or the tasks we require you to solve
- That you will *take pride in your work, and your learning*
- That you will *strive to build friendships and collaborations* with your colleagues

What you can expect from us

- That we will provide you with a thoroughly well thought-out, consistent, organized course designed to prepare you extremely well for your career
- That we will provide you with different opportunities to learn the concepts the course is about
- That we will help you develop your skills as a computer scientist
- That we will treat you with respect, patience, and understanding
- That we will evaluate your work fairly in the context of what skills and knowledge you need in order to succeed in your career
- That we will help you establish an environment that is good for learning, for spending your time in, for building your career
- That we will respond quickly and effectively to any issues that may arise during the course that could impact your learning

Why should we put effort into the non-academic aspects of our career?

- Because it will mean UTSC will become a place you feel comfortable in, while you're here
- Because it will provide you with support from your colleagues, every day, for every course and problem you may find
- Because it will help you build a group of people who'll be a good part of your life in the future
- Because we want to make sure the world sees you for who you are
- *Because the environment you build now will carry onto your professional life*

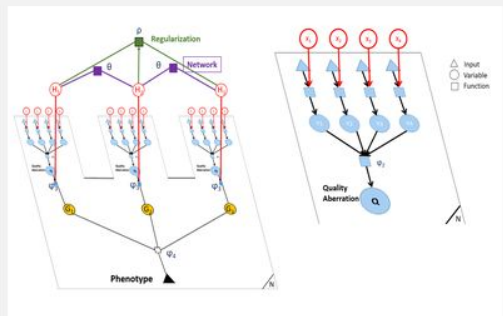
What do people in Computer Science look like?



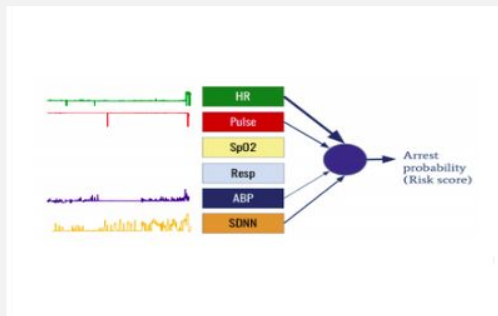
Here are few very cool people working



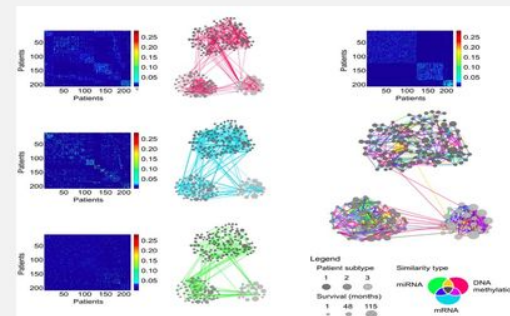
Cool things they do!



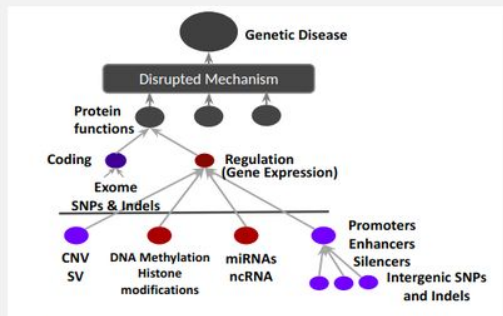
Graphical models for variant aggregation



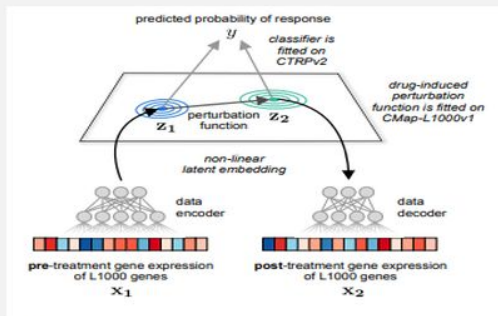
Cardiac Arrest Prediction



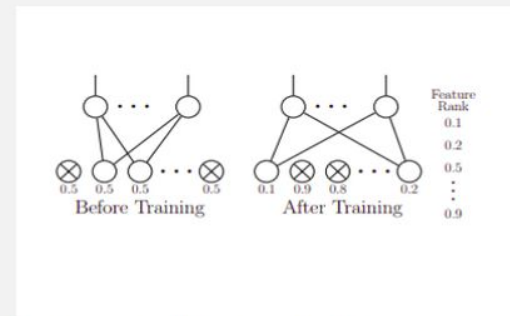
Similarity Network Fusion



Patient Heterogeneity



Drug Response Prediction



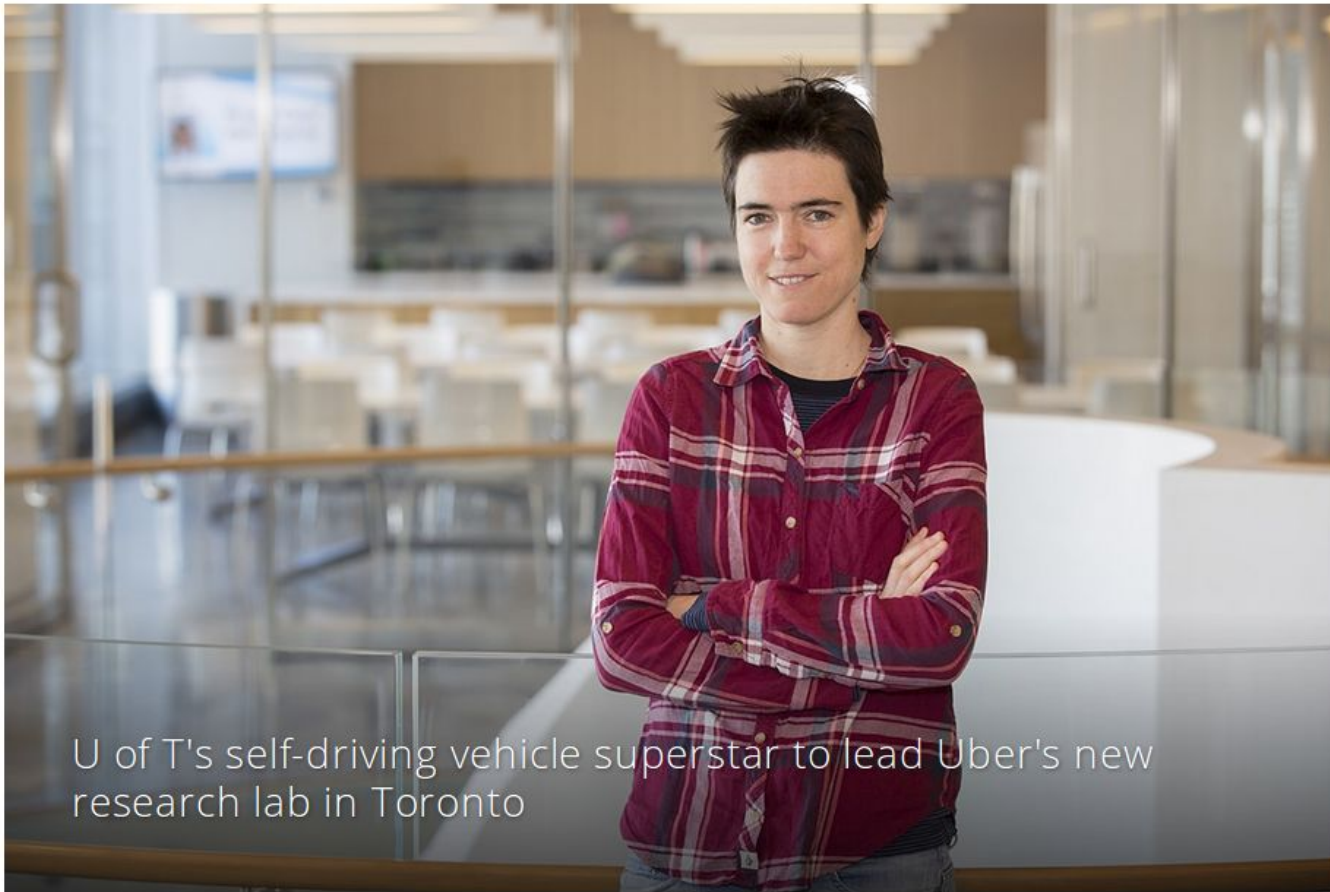
Dropout Feature Ranking of Deep Learning Models

Here are few very cool people working with CS

Raquel Urtasun and Sanja Fidler, along with colleagues in Spain, designed an algorithm that analyzes a person's photograph to determine whether the wearer's outfit is stylish. It also suggests ways to improve the ensemble and the subject's overall appeal.



University of Toronto researchers Raquel Urtasun, left, and Sanja Fidler are creating an app that assesses clothing and recommends how to be more fashionable. (STEVE RUSSELL / TORONTO STAR)



U of T's self-driving vehicle superstar to lead Uber's new research lab in Toronto

U of T Associate Professor Raquel Urtasun is joining Uber's new branch for Advanced Technologies Group (photo by Johnny Guatto)

May 08, 2017

NVIDIA Opens AI Research Lab in Toronto

 June 14, 2018 by [staff](#)  [Leave a Comment](#) 

Over at the [NVIDIA Blog](#), Gavriel State writes that the company has opened a new AI Research Lab in Toronto. The AI Research Lab will be led by computer scientist and University of Toronto professor [Sanja Fidler](#).

“With the new lab, our goal is to triple the number of AI and deep learning researchers working there by year's end. It will be a state-of-art facility for AI talent to work in and will expand the footprint of our office by about half to accommodate the influx of talent.

Fidler's main research interests are deep learning and computer vision, with connections to natural language processing. While taking up the helm as director of AI research for the NVIDIA Toronto office, she'll continue in her role as an assistant professor in the department of computer science at the University of Toronto.

The move comes as the Canadian government is touting the opening of domestic offices by top-tier tech multinationals.



Sanja Fidler



Borealis AI increases brain power by adding top AI talent

Curiosity-driven research dedicated to achieving state-of-the-art in machine learning and artificial intelligence.

Borealis AI Lab in downtown



So, is it worth knowing more about CS?

- **Absolutely!** It's interesting, it's challenging, it's in demand, it opens doors to well paid jobs!
- You can use it to change the way you live, **to make the lives of others better**
- You can **bring your hobbies** into it!

The Art



The Artists
-Harold Cohen and AARON



You can also use it to create music, art, images, movies, make beautiful things



What is new for this Summer?

- We will meet **in-person** for weekly lectures
- All exams (midterm and final) will be **in-person** as well!

Meeting Section	Day(s)	Start(s)	End	Location	Current	Max	Wait	Delivery Mode
LEC01	MO	14:00	15:00	SW 319	175	175	Y	In-person
	WE	13:00	15:00	SW 309	175	175	Y	In-person

Office Hours

This week: Monday 3 to 4 IC 484

From next week: Wednesday 3 to 4

Getting help!

- Attend lectures, tutorials, office hours.
- Google your **compiler error** messages
- Ask on Piazza
 - **Do's**
 - Be polite and professional
 - Be helpful to fellow students
 - Post any code **covered in class**
 - Question about advanced topic - Sure! Ask away!! But please tag the question "off topic"
 - In case of doubt, Write the **compiler error message** and tell us what google told you (make a private post)
 - **Don'ts**
 - Do not ask for solutions to problems, exercises, assignments, or quizzes
 - Avoid vague questions like "my code doesn't work..Help!"
 - Don't post your code

Grading Policy

Item	Notes	Value
Weekly Exercises (auto-tested)	Develop problem solving ability.	5% (Only your best 9 count)
Weekly Questionnaires (not marked)	Help you reflect on your learning, and provide feedback	5%
Assignments (×3)	Automarked	15%
Tutorial Exercises	Not marked	5%
Midterm Test	Date TBA	25%
Final Exam	Date TBA	45%

Things to do this week

- If you haven't done so: **read carefully the course information handout**
- Download and carefully **go through the first set of notes (upto Page 10) – before lecture**
- Set up a working compiler
- There is no tutorial and there are no practical sessions for week 1
- Make sure you are **enrolled in the Piazza forum** (look in your utoronto email address for an email from us)
- Complete the **entry questionnaire (on Quercus) and the first weekly exercise.**