Why did the computer get glasses?

To improve its web sight!

LearnFunnyJokes.com
Welcome to CMPT 120

Introduction to Computing Science and Programming 1

Anne Lavergne
Today’s Menu

• Introducing the course
  • What is this course all about?
  • What kind of students take CMPT 120?
  • What do we need for this course?
  • How does this course work?

• Q&A
  • Waiting list

• Let’s get started!
  • How does a computer work?
  • Python demo

• Summary

• Next lecture!
What is this course all about?

- Title: “Introduction to Computing Science and Programming 1”

  Learn fundamental concepts of Computing Science e.g. problem solving, algorithms, and get a sneak preview of some of its most interesting topics such as big data and graphics 😊

  Learn software development using the programming language Python.
What is this course all about?

• Title: “Introduction to Computing Science and Programming 1”

Learn software development using the programming language Python

Learn fundamental concepts of Computing Science e.g. problem solving, and get a sneak preview of some of its most interesting topics like big data and graphics 😊
From the movie

Source: https://www.youtube.com/watch?v=XiwbkyjrmQ
Why **Computing Science** and not **Computer Science**?

- Title: “Introduction to **Computing Science** and Programming 1”

**Computing:** the systematic study of computation.

Wiki says that **computation** is any type of calculation that is well-defined, i.e., precise and described as a step-by-step process.
Why Computing Science and not Computer Science?

• From Edsger Dijkstra: “Computing science is as much about computers as astronomy is about telescopes.”

• From wiki: “Despite its name, a significant amount of computer science does not involve the study of computers themselves.”

• Computer: a tool
• Computing: field of study
What is **Computing Science**?

Software development and software engineering, databases, operating systems, computer graphics, multimedia systems, data science, machine learning, programming languages, computability and complexity theory, cryptography, networking, human-computer interaction, robotics, …

---

**CMPT 340 - Biomedical Computing (3)**

The principles involved in using computers for data acquisition, real-time processing, pattern recognition and experimental control in biology and medicine will be developed. The use of large data bases and simulation will be explored. Prerequisite: Completion of 60 units including one of CMPT 125, 126, 128, 135, with a minimum grade of C- or CMPT 102 with a grade of B- or higher.

---

**CMPT 353 - Computational Data Science (3)**

Basic concepts and programming tools for handling and processing data. Includes data acquisition, cleaning data sources, application of machine learning techniques and data analysis techniques, large-scale computation on a computing cluster. Prerequisites: CMPT 235 and [STAT 150 or CMPT 240] or [CMPT 270 or CMPT 271, ENSC 280, or MIE 210], with a minimum grade of C-.

---

**CMPT 354 - Database Systems I (3)**

Logical representations of data records. Data models. Studies of some popular file and database systems. Documentation. Other related issues such as database administration, data dictionary and security. Prerequisites: CMPT 235 and [MACM 101 or ENSC 251 and ENSC 252], all with a minimum grade of C-.

---

**CMPT 361 - Introduction to Visual Computing (3)**

Provides an introduction to the fundamentals of computer graphics and computer vision (visual computing). Topics include graphics pipelines, sampling and aliasing, geometric transformations, projection and camera models, shading, texturing, color theory, image filtering and registration, shading and illumination, raytracing, rasterization, animation, optical flow, and game engines. Prerequisites: CMPT 225 and MATH 232 or 240, all with a minimum grade of C-.

---

**CMPT 362 - Mobile Applications Programming and Design (3)**

Teaches students how to design and implement smartphone applications. Topics include development environment, phone simulator, key programming paradigms, UI design including layouts, fragments, and activities, data persistence, threads, services, embedded sensors, and location based services (e.g., Google Maps). Concepts are reinforced through programming assignments and group projects. Prerequisite: CMPT 225 with a minimum grade of C-. Students with credit for CMPT 359 may not take this course for further credit.

---

**CMPT 363 - User Interface Design (3)**

This course provides a comprehensive study of user interface design. Topics include: goals and principles of UI design (systems engineering and human factors), historical perspective, current paradigms (widget-based, natural model, graphic design, ergonomics, metaphor, constructivist/behaviorist approach, and visual languages) and their evaluation, existing tools and packages (dialogue models, event-based systems, prototyping), future paradigms, and the social impact of UI. Prerequisite: CMPT 225 with a minimum grade of C-.

---

**CMPT 365 - Multimedia Systems (3)**

Multimedia systems design, multimedia hardware and software, issues in effectively representing, processing, and retrieving multimedia data such as text, graphics, sound and music, image and video. Prerequisite: CMPT 225 with a minimum grade of C-.

---

**CMPT 371 - Data Communications and Networking (3)**

Data communication fundamentals (data types, rates, and transmission media). Network architectures for local and wide areas. Communications protocols suitable for various architectures. ISO protocols and internetworking. Performance analysis under various loadings and channel error rates. Prerequisites: CMPT 225 and [MATH 151 or MATH 150] or [MATH 154 or MATH 157] with a grade of at least B+ may be substituted for MATH 151 (MATH 150).

---

Source: Computing Science - Fall 2023 Calendar - Simon Fraser University (sfu.ca)
What kind of students take CMPT 120?

- Students from a variety of faculties and departments
  - Let’s have fun! Let’s see from which SFU departments we are:
    - Join at menti.com use code 3148 8874
- Students with no previous software development experience
What do we need for this course?

- Access to our course web site: 
  https://www.cs.sfu.ca/CourseCentral/120/alavergn/index.html
- Access to CourSys
- Python
  - Python IDLE: Integrated Development and Learning Environment for Python
  - See the link called Install Python 3.12.1 (or latest) IDLE under Week 1 on our course web site
Why Python?

- Simple and easy to use (easy syntax)
- Yet powerful
- Readable code (i.e., English-like code)
Why Python?

From PYPL Popularity of Programming Language:

<table>
<thead>
<tr>
<th>Rank</th>
<th>Change</th>
<th>Language</th>
<th>Share</th>
<th>1-year trend</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>Python</td>
<td>28.2 %</td>
<td>+0.5 %</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>Java</td>
<td>15.73 %</td>
<td>-0.9 %</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>JavaScript</td>
<td>8.91 %</td>
<td>-0.6 %</td>
</tr>
<tr>
<td>4</td>
<td>↑</td>
<td>C/C++</td>
<td>6.8 %</td>
<td>-0.0 %</td>
</tr>
<tr>
<td>5</td>
<td>↓</td>
<td>C#</td>
<td>6.67 %</td>
<td>-0.3 %</td>
</tr>
<tr>
<td>6</td>
<td>↑</td>
<td>R</td>
<td>4.59 %</td>
<td>+0.6 %</td>
</tr>
<tr>
<td>7</td>
<td>↓</td>
<td>PHP</td>
<td>4.54 %</td>
<td>-0.7 %</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>TypeScript</td>
<td>2.92 %</td>
<td>+0.2 %</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td>Swift</td>
<td>2.77 %</td>
<td>+0.6 %</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td>Objective-C</td>
<td>2.34 %</td>
<td>+0.2 %</td>
</tr>
</tbody>
</table>

Source: https://pypl.github.io/PYPL.html
Why Python?

From Northeastern University:

Top 10 Most In-Demand Programming Languages

The 10 most in-demand programming languages in 2023 are:

1) Structured Language Query (SQL)
2) Python
3) Java
4) JavaScript
5) C#
6) C++
7) R
8) C
9) Go
10) Perl

Source: https://graduate.northeastern.edu/resources/most-popular-programming-languages/
Why Python?

From statista.com:

Most used programming languages among developers worldwide as of 2023

- JavaScript: 63.61%
- HTML/CSS: 52.97%
- Python: 49.28%
- SQL: 48.66%
- TypeScript: 38.87%
- Bash/Shell (all shells): 32.37%
- Java: 30.55%
- C#: 27.62%
- C++: 22.42%
- C: 19.34%
- PHP: 18.58%
- PowerShell: 15.59%
- Go: 13.24%

How does this course work?

• Let’s have a look at our course web site and CourSys
Q&A: Any Questions?

• Waiting list
  • Hang in there! 😊
    • Continue to come to lectures
    • No grade-based activities due during Week 1 and Week 2
  • Waiting lists will be resolved by the end of Week 1
    • If you have not been enrolled into CMPT 120 by the start of Week 2, you may wish to try to enroll next semester.
  • Please, do not send emails to the instructor 😞
• Good luck!
Let’s get started!

- How does a computer work?
Python Demo

- Let’s write a little Python program and see how the computer executes it.
Summary

✓ Introducing the course
  ✓ What is this course all about?
  ✓ What kind of students take CMPT 120?
  ✓ What do we need for this course?
  ✓ How does this course work?

✓ Q&A
  ✓ Waiting list

✓ Let’s get started!
  ✓ How does a computer work?
  ✓ Python demo
How to prepare ourselves for the next lecture?

- Become familiar with our course web site
- Print/download and read the Lecture Notes for next lecture
- Read the Readings
- Download and install Python IDLE
- Bring your questions ... and bring your laptop!!!
Next Lecture

- Software development process
  - Natural and Formal languages
  - Algorithm
  - Our first program
    - Execution flow
    - Interpreted program versus compiled
  - Comments
  - Python
    - print function
    - strings
    - variables

How best to prepare for our next lecture?
- Become familiar with our course web site
- Download/print the Lecture Notes for next lecture
- Read next lecture’s readings
- Install Python
- Bring questions … and your laptop!!!

Source: https://www.dreamstime.com/