Problem-Solving and Abstraction

30 August 2021
What will this course be about?
Consider: Two graphs about e-commerce
What do you notice?

What do you wonder?

Source: The New York Times
Net change in retail jobs over the last 15 years

Source: The New York Times
What does any of this have to do with this class?
We’ll cover a lot of what data scientists do.
But this is an introduction to computer science, not data science.
Understanding what processes can be automated and how to organize the information those processes depend on is part of computer science – specifically, this is what programming is all about.
Understanding how to sanity check and test your analyses so that you trust them is important in both data science and computer science.
This course will focus on computer science techniques that matter for dealing with data and information, as mediated through the task of programming.
When we say “data”, we don’t just mean graphs, spreadsheets, and statistics.

We’ll also be looking at the information and data that underlie applications in various domains.
Consider: GMail
Blaine, me  Projects  Recent project updates — Key highlights: the team has started on the ke...  2:25 PM
Clarence .. Vijay .. 13  RSVP for team lunch and bike ride! — We’re pleased to announce that we will have...  1:35 PM
Brianna, John 2  Tickets  Ticket request #610007 has been approved! — Your ticket has been appro...  12:25 PM
William, me 4  Support Archive  Thank you for your inquiry — We have received your message and ...  April 17
Hilton Honors  Your Upcoming Reservation #20983746 — Tim Smith, thank you for choosing Hilton...  April 17
Virgin Atlantic  Confirmation for Flight VA2345 SFO to NYC — Wednesday, November 7th 2015, San...  April 17
Jack  FW: What “the future of innovation” Looks Like — A good read! Highly recommende...  April 17
Xander  Photos from my road trip — Hi all, here are some highlights from my vacation. What ...  April 16
Richard, Matthew, me 3  Product Strategy classes — He emailed me about his latest work. Here’s what we rev...  April 16
Peter, Shalini 2  Business trip — Hi, I made a reservation for the hotel you talked about. It looks fan...  April 16
Roy, Alex, John Jose 5  Book  Book you recommended — About to go on a trop and was hoping to learn mo...  April 16
<table>
<thead>
<tr>
<th>Subject</th>
<th>Body</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blaine, me</td>
<td>Projects Recent project updates — Key highlights: the team has started on the ke... 2:25 PM</td>
</tr>
<tr>
<td>Clarence .. Vijay .. 13</td>
<td>RSVP for team lunch and bike ride! — We’re pleased to announce that we will have... 1:35 PM</td>
</tr>
<tr>
<td>Brianna, John 2</td>
<td>Tickets Ticket request #610007 has been approved! — Your ticket has been appro... 12:25 PM</td>
</tr>
<tr>
<td>William, me 4</td>
<td>Support Archive Thank you for your inquiry — We have received your message and... April 17</td>
</tr>
<tr>
<td>Hilton Honors</td>
<td>Your Upcoming Reservation #20983746 — Tim Smith, thank you for choosing Hilton... April 17</td>
</tr>
<tr>
<td>Virgin Atlantic</td>
<td>Confirmation for Flight VA2345 SFO to NYC — Wednesday, November 7th 2015, San... April 17</td>
</tr>
<tr>
<td>Jack</td>
<td>FW: What “the future of innovation” Looks Like — A good read! Highly recommende... April 17</td>
</tr>
<tr>
<td>Xander</td>
<td>Photos from my road trip — Hi all, here are some highlights from my vacation. What... April 16</td>
</tr>
<tr>
<td>Richard, Matthew, me 3</td>
<td>Product Strategy classes — He emailed me about his latest work. Here’s what we rev... April 16</td>
</tr>
<tr>
<td>Peter, Shalini 2</td>
<td>Business trip — Hi, I made a reservation for the hotel you talked about. It looks fan... April 16</td>
</tr>
<tr>
<td>Roy, Alex, John Jose 5</td>
<td>Book Book you recommended — About to go on a trop and was hoping to learn mo... April 16</td>
</tr>
</tbody>
</table>
user-design
security
privacy
machine maintenance
accessibility
data management
machine learning
...
...
Computer science + X
GMail requires very different kinds of data than the job statistics, but we face some similar problems:

We need to organize it, store it, present it.
What will we do in this course?
1. Identify and organize the data needed to solve the problem

2. Break the problem down into subproblems that can be solved with computations

3. Express computations over the data

4. Test those computations to make sure they’re doing what they’re supposed to

Data design

Programming

Programming/CS

Testing
Think about whether it’s a good idea to solve the problem, and how your solution might affect the world around you.
This course teaches skills that will help you in computer science and beyond.
Three major units:

Tabular / data science data
Other core CS data structures and how to program with them
Additional data structures and programming techniques
Three major units:

- Tabular / data science data
- Other core CS data structures and how to program with them
- Additional data structures and programming techniques

<table>
<thead>
<tr>
<th>Pyret</th>
</tr>
</thead>
<tbody>
<tr>
<td>Python</td>
</tr>
</tbody>
</table>
Goals

Apply fundamental data-organizations (called data structures) to capture the information in a computing problem.

Break down a computing question into manageable smaller problems.

Write programs to compute answers to questions over fundamental data structures.

Check whether your programs behave as intended/required.
CMPU 101 Student information

Fall 2021
* Required

Name on AskBanner:
Your answer

Preferred name (optional):
Your answer

Pronouns (optional):
Your answer

Enrollment:
- [ ] Registered
- [ ] Registered but planning to drop

forms.gle/Km5KTtYfXgH8geZq7
Lectures:

Monday & Wednesday, 1:30–2:45 p.m.
Sanders Classroom 006

Lab:

Friday, 3:10 p.m.–5:10 p.m.
Sanders Classroom 006
CMPU 101 § 1
Problem-Solving and Abstraction
Fall 2021

Monday 1:30–2:45 p.m.
Wednesday 1:30–2:45 p.m.
Friday 3:10–5:20 p.m.
Sanders Classroom 006

Prof. Jonathan Gordon

<table>
<thead>
<tr>
<th></th>
<th>Monday</th>
<th>Wednesday</th>
<th>Friday</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Read Syllabus</td>
<td>Lecture 1</td>
<td>Lab 1</td>
</tr>
<tr>
<td></td>
<td>Read 1 Overview</td>
<td>Our first programs</td>
<td>Getting started</td>
</tr>
<tr>
<td></td>
<td>Read 3 Getting Started</td>
<td>Lecture 2</td>
<td>Names, types, and organization</td>
</tr>
<tr>
<td></td>
<td>Read 4 Naming Values</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

[cs.vassar.edu/~cs101/1]
CMPU 101 §1
Problem-Solving and Abstraction
Fall 2021

Monday 1:30–2:45 p.m.
Wednesday 1:30–2:45 p.m.
Friday 3:10–5:10 p.m.
Sanders Classroom 006

Prof. Jonathan Gordon
cs.vassar.edu/~cs101/1

Overview

This course introduces fundamental concepts of computer science. Its major goal is to introduce students to the principles of systematic problem-solving through programming and the basic rules of computation. We will explore the art and science of problem-solving using the computer as a tool. We will write programs in Pyret, a simple yet powerful student learning language, and Python, a popular language for work across computer science and data science.
CMPU 101 Fall 2021

**DESCRIPTION**
Edit your course description on the Course Settings page.

**THINGS TO DO**
1. Add students or staff to your course from the Roster page.
2. Create your first assignment from the Assignments page.

<table>
<thead>
<tr>
<th>ACTIVE ASSIGNMENTS</th>
<th>RELEASED</th>
<th>DUE (EDT)</th>
<th>SUBMISSIONS</th>
<th>% GRADED</th>
<th>PUBLISHED</th>
<th>REGRADES</th>
</tr>
</thead>
</table>

You currently have no assignments.
Create an assignment to get started.

Create Assignment
Grading

Weekly lab exercises 20%

Labs
Assignments
Exam 1
Exam 2
Final
Grading

Approximately weekly programming homework assignments
Grading

- Two midterm exams: 20% each
- Labs: 20%
- Assignments: 20%
- Exam 1: 20%
- Exam 2: 20%
- Final: 20%

*Two midterm exams*
Grading

Regularly scheduled final exam
“All through our education, we are being taught a kind of reverse mindfulness. A kind of Future Studies where – via the guise of mathematics, or literature, or history, or computer programming, or French – we are being taught to think of a time different to the time we are in. Exam time. Job time. When-we-are-grown-up time.

To see the act of learning as something not for its own sake but because of what it will get you reduces the wonder of humanity. We are thinking, feeling, art-making, knowledge-hungry, marvelous animals, who understand ourselves and our world through the act of learning. It is an end in itself. It has far more to offer than the things it lets us write on application forms. It is a way to love living right now.”

Matt Haig, Notes on a Nervous Planet
All questions have been resolved
Enjoy the rest of your day
Starting to program
What do you notice?

What do you wonder?
We're trying to make sense of the problem.

We start with the *data* before we dive in to try to *do* it.
We might want to compute the heights of the stripes from the overall flag dimensions, which means we need to write programs over *numbers*.

We need a way to describe *colors* to our program.

We need a way to create images based on simple *shapes* of different colors.
We might want to compute the heights of the stripes from the overall flag dimensions, which means we need to write programs over *numbers*.

We need a way to describe *colors* to our program.

We need a way to create images based on simple *shapes* of different colors.
Look at an interaction like

\[ > \ (3 \ + \ 4) \ast \ (5 \ + \ 1) \]

42
Look at an interaction like

```
> (3 + 4) * (5 + 1)
42
```

*Expression*: a computation that produces an answer (it may be part of another computation)
Look at an interaction like

\[ > (3 + 4) \times (5 + 1) \]

Value: a computation that can’t be computed further (it is its own answer)

Expression: a computation that produces an answer (it may be part of another computation)
Look at an interaction like

> \((3 + 4) \times (5 + 1)\)

42

*Expression*: a computation that produces an answer (it may be part of another computation)

*Value*: a computation that can’t be computed further (it is its own answer)

*Program*: a sequence of computations that you want to run
We might want to compute the heights of the stripes from the overall flag dimensions, which means we need to write programs over *numbers*.

We need a way to describe *colors* to our program.

We need a way to create images based on simple *shapes* of different colors.
Names can be given as strings, e.g., "blue"
We might want to compute the heights of the stripes from the overall flag dimensions, which means we need to write programs over *numbers*.

We need a way to describe *colors* to our program.

We need a way to create images based on simple *shapes* of different colors.
> include image

> circle(50, "solid", "red")
Hello, computer
We use computers every day as electronic black boxes that do amazing things by collecting, storing, retrieving, and transforming data.
Phones, tablets, desktops, watches, thermostats, medical devices, etc. are all computers.

*Computers are devices that carry out the operations of a computation.*
A *computation* is a series of numerical calculations and symbol manipulations.

Numerical calculations include:

- Basic arithmetic operations (add, subtract, multiply, divide)
- Basic trigonometric functions (sine, cosine, tangent)

Symbolic manipulations include:

- Logical comparison of numbers or symbols
- Decisions of what instructions to do next
- Substitutions of one string of letters and numbers for another
Amazing computations can be carried out when trillions of such simple operations are arranged in the proper order, e.g.,

- forecasting tomorrow’s weather
- deciding where to drill for oil
- designing the wings of an aircraft with enough lift to fly
- finding which physical places are most likely to be visited by a person calling for a taxi
- figuring out which two people would make a great couple 😍😘
How do you draw so well?

Practice.

It must be an innate gift...
A gift from God...

It's practice.

I'll never understand how some people are so talented...
A mystery...

Practice.

© Sarah Andersen
We’ve got a big journey ahead of us. I hope you’re excited!
Acknowledgments

This lecture incorporates material from:

Kathi Fisler, Brown University
Doug Woos, Brown University