On your way in...

Pick-up:
1. Course Syllabus print-out
2. Homework 00 print-out

(There will ~always be something to grab on your way in!)
BEFORE WE BEGIN...
I’m Iris Howley. She/Her/Hers. You can call me “Iris” or “Professor Howley”, whichever you’re more comfortable with! I’m your professor for this lecture section.

I’m also out of town at a conference on Artificial Intelligence, Ethics, & Society with a couple of the students who do research with me.
Prof. Jeannie Albrecht has very kindly volunteered to handle this first class session for me. She should be able to answer most of your questions, and you can email me at iris@cs.williams.edu!

I’ll be back on Monday to answer more questions!
Please fill-out the attendance sheet that’s being passed around the class. (If you don’t, you might get dropped from the class.) **Find the attendance sheet before you leave!**

If your name isn’t on there, please write it in one of the blank spaces, along with your Williams Unix ID so we can add you to the waitlist.
If you’re interested in switching lab sections (or lecture sections), you can do this if you find another student to swap places with you.

And then email me, cc’ing the student you’re swapping with, so we can make the change.

Please do this prior to Monday!
If you’re considering dropping this class, please do so as soon as possible.

There’s lots of people on the waitlist, and we won’t be able to get them in the class if you wait until Tuesday to drop!
Welcome to CS 134!

Introduction to Computer Science

Iris Howley
She/Her/Hers

Spring 2020
Why are you interested in computer science?

Take 2 minutes to write down all the reasons you enrolled in this course.
Welcome to Computer Science!

Take 2 more minutes and...

....introduce yourself to your neighbors

....find 2 more reasons, different from your own, for taking this course.
What are some reasons you are interested in computer science?

Share with the class!
Why I’m a computer scientist...
Work by Lester Lee ‘19

1972 OLYMPIC POSTER

RONALD B. KITAJ, 1970

Medium
color silkscreen on paper

Dimensions
34.09 x 25.16 in. (86.6 x 63.9 cm)

Description
Description not available.

Info Viz collaboration w WCMA
Work by Michaela Smith ‘20

(FROM A "RASIKAPIYA" SERIES)

Maker: Ibrahim
Medium: opaque watercolor on paper
Creation Date: 1692
Accession Date: 1980-01-01

Info Viz collaboration w WCMA
Work by Tongyu Zhou ‘20 and Kelvin Tejeda ‘20

Teaching people AI in summer research
1. Find two different parameter combinations that will result in mastery if the student answers correctly. *Hint: make P(learned if correct) ≥ 0.95 and press "answer correct" to verify your results.*

2. What happens to P(learned if correct) and P(learned if wrong) if P(guess) and/or P(slip) exceeds 0.5?

3. What happens to P(learned) if the student answers incorrectly? *Hint: compare P(learned if wrong) with P(known) (aka. your previous P(learned)).*

4. Keep exploring! Can you find any other flaws or interesting characteristics of BKT?

*Hint: hover over the parameters to see how they impact the simulator.*

Teaching people AI in summer research.
COMPUTER SCIENCE IS GREAT FOR MAKING.

COMPUTER SCIENCE IS GREAT FOR

PROBLEM SOLVING.

COMPUTER SCIENCE IS GREAT FOR

UNDERSTANDING PEOPLE

...AND IT'S FUN!
It’s almost lunchtime...

• ...let’s make a sandwich

• What are the steps to making a peanut butter & jelly sandwich?

• (What’s the first step?)
Peanut Butter Sandwich Instructions

• [https://youtu.be/cDA3_5982h8?t=37](https://youtu.be/cDA3_5982h8?t=37)
  • (Watch until 1:35 when the daughter appears)
The sandwich-making instructions are an algorithm for making a peanut butter sandwich. They need to be pretty specific for a computer to understand.

Computer scientists design algorithms to solve entire sets of problems. In this class, we learn to design and apply algorithms.

Why did we just do that?
Who was the algorithm designed for?

Was it designed for people who like a lot of jelly or a little?

What about people allergic to peanuts?

As computer scientists, we make decisions when we design algorithms.

These decisions can impact a lot of outcomes, including the people that use our systems.
In this course (according to our course description):

“Students will learn to design algorithms to search, sort, and manipulate data in application areas like text and image processing, social networks, scientific computing, and databases.”

We’re learning a programming language called ‘python’ in this course.

But programming languages are just a way for computer scientists to communicate algorithms → how we solve problems.
We’ll use other tools that computer scientists use

- Terminal: command line interface for the operating system
- git: version control, great for collaboration
- GitLab: a web interface built on git
- Atom: a text editor (like a Word Processor, but simpler)
CS 134: Introduction to Computer Science
Who We Are

Iris Howley
iris@cs.williams.edu
Co-instructor

Shikha Singh
shikha@cs.williams.edu
Co-instructor

Lida Doret
lida@cs.williams.edu
Lab Instructor
Who We Are

Harun Curak
Diego Esparza
Nathan Thimothe
Maria Chapman
Amelia Chen
Caleb Dittmar
Hugo Hua
Brian Kamau
Sarah Lyell
Yash Mangal
Rachel Nguyen
Minh Phan
Alicia Smith Reina
Mira Sneirson
Jules Walzer-Goldfeld
Emma Wuerth
Why is my lab instructor different from my lecturer?

TWO SECTIONS, ONE CS134!
Important Resources
The Course Website

“No, we don’t use Glow”


### CSCI 134 - Spring 2020
**Introduction to Computer Science**

#### Home
- **Section:** CSCI 134-02, 134-08, 134-09
- **Instructor:** Shikha Singh
- **Email:** shikha@cs.williams.edu
- **Phone:** x4773
- **Office:** TBL 308
- **Office Hours:** Mondays: 2:30-4 pm, Wednesdays: 12-2:00 pm & by appointment
- **Lectures:** MWF: 9:00-9:50 am, Schow Science Library 030a
- **Labs:** Tuesday 1-2:30 pm (Section 08) & 2:30-4:00 pm (Section 09) in TCL 217a
  
  Labs are due Thursday @ 11 pm

#### Section
- **Section:** CSCI 134-03, 134-04, 134-05
- **Instructor:** Iris Honley
- **Email:** iris@cs.williams.edu
- **Phone:** x4663
- **Office:** TCL 308
- **Office Hours:** Wednesdays 12-1:00 pm, Thursdays TBD & by appointment
- **Lectures:** MWF: 11:00-11:50 am, Schow Science Library 030a
- **Labs:** Monday 1-2:30 pm (Section 04) & 2:30-4:00 pm (Section 05) in TCL 217a
  
  Labs are due Wednesday @ 11 pm

#### TAs
- Harun Cural, Diego Esparza, Nathan Thimmothe, Maria Chapman, Amelia Chen, Caleb Dittrar, Hugo Hua, Brian Kuman, Sarah Lynch, Yash Mangal, Rachel Nguyen, Miah Phan, Mira Scuzzio, Jules Walter-Godfeld, Emma Wuerth

Now with:
- The syllabus!
- Office hours!
- TA Hours!
- The textbook!
- Lecture notes!
- Homeworks!
- The labs!
- Contact info!
The Textbook


• We use this textbook the first half of the course
• Chapters are assigned on the course schedule/syllabus
Syllabus
<table>
<thead>
<tr>
<th>Week of</th>
<th>Monday</th>
<th>LAB</th>
<th>Wednesday</th>
<th>Friday</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feb. 3</td>
<td>—</td>
<td>I. PYTHON AND GITLAB</td>
<td>—</td>
<td>1. Hello, world! (TP1)</td>
</tr>
<tr>
<td>Feb. 10</td>
<td>2. Expressions (TP2)</td>
<td>II. PROCEDURE</td>
<td>3. Functions (TP3)</td>
<td>Winter Carnival</td>
</tr>
<tr>
<td>Feb. 24</td>
<td>7. Strings (TP8-9)</td>
<td>IV. FACULTY TRIVIA</td>
<td>8. Lists, Tuples (TP10,12)</td>
<td>9. Files (TP14)</td>
</tr>
<tr>
<td>M. 22&amp;29</td>
<td>Spring Break</td>
<td>Spring Break</td>
<td>Spring Break</td>
<td>Spring Break</td>
</tr>
<tr>
<td>Apr. 6</td>
<td>19. Images</td>
<td>VII. IMAGES</td>
<td>20. Slack</td>
<td>21. Multiple Classes</td>
</tr>
<tr>
<td>Apr. 27</td>
<td>* Slack</td>
<td>IX. RECURSIVE TREES</td>
<td>28. Object Persistence</td>
<td>29. Scope</td>
</tr>
<tr>
<td>May 11</td>
<td>33. Special Topics</td>
<td>X. PROJECT (CONT.)</td>
<td>34. Special Topics</td>
<td>35. Evaluations</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Final Exam</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Workload

- Readings from textbook, as on the schedule
- Weekly written homeworks
- Weekly lab assignments

Super useful: Iris & Shikha’s Student Help hours
TA Student Help hours

- Midterm exam on Tuesday, March 12 (evening)
- Scheduled Final exam
Student Help Hours
Elsewhere called “Office Hours”

• TA Student-Help Hours
  ▪ Sun. 6:30-10p, Mon 7-9:30p, Tue 7:30-10p, Wed 6-9:30p, Thu 6-10p

• Iris Student-Help Hours (TCL 308)
  ▪ Wednesday 12-1p
  ▪ Thursday 10a-12p

• Shikha Student-Help Hours
  ▪ Monday 2:30-4pm (TBL 309b)
  ▪ Wednesday 12-2p (CS Common Room (outside TCL308))

• Lida Student-Help Hours (TCL205)
  ▪ Wednesday 1:30-3pm

Check the website for updated office hours! http://www.cs.williams.edu/~cs134/
Grades

• **Homeworks (15%)**
  - Short answer programming & problem solving questions
  - Due Mondays in class (generally)
  - Use pencil and paper

• **Labs (40%)**
  - Meet Mon/Tues for a 1.5 hour block
  - Monday labs due Wed @ 11pm | Tuesday labs due Thurs @ 11 pm
  - Review lab as soon as it comes out (~Fridays)

• **Midterm (20%)**
  - Evening exam on Thursday, March 12.

• **Final (25%)**
  - Scheduled Final Exam

• **Participation +/- 5%**
Class Participation

I like interaction in my classes!

• Many ways to participate:
  ▪ Ask questions! (there are no bad questions in my class)
  ▪ Answer questions (there are no wrong answers in my class)
  ▪ Talk to me after class/office hours

• Help create a vibrant, positive and inclusive classroom environment!

We will talk much more about this on Monday, but my classes are generally more than half small group work. Hands-on learning is extremely effective!
This is a fast-paced course!

• How to succeed:
  ▪ Do readings before class
  ▪ Read and think about labs as soon as they are released
  ▪ Seek help! Use resources!

• Learning to program is all about PRACTICE, PRACTICE, PRACTICE!
  ▪ Just like learning a musical instrument, learning to ski, or building muscle, it requires repetition and dedication
  ▪ Can’t passively absorb material
  ▪ Don’t be afraid to fail and make mistakes—in fact you are encouraged to do so!
  ▪ No one learns anything without making mistakes and learning why and how to fix them
Students Who Need Accommodations

• If formal accommodations need to be made to meet your specific learning or physical abilities, please contact one of us as soon as possible to discuss appropriate accommodations. Please also contact the Director of Accessible Education, Dr. G. L. Wallace (413-597-4672) or the Dean’s office (413-597-4171). We will work together to ensure this class is as accessible and inclusive as possible.

• Also, students experiencing mental or physical health challenges that are significantly affecting their academic work are encouraged to contact one of us or to speak with a dean. The deans can be reached at 413-597-4171.
Honor Code
Departmental Honor Code & Computer Usage Policy


- People learn best by doing!
- Do your own work!
Honor Code Guideline

• “Any work that is not your own is considered a violation of the Honor Code.”

• “Help locating errors and interpreting error messages are allowed, but a student may only receive help in correcting errors of syntax; help in correcting errors of logic is strictly forbidden.”

• “In general, if you are taking photos of someone else’s screen, looking at someone else’s screen, or telling someone else what to type, it is likely your/their work is no longer the work of an individual student.”
Honor Code

• Resources to consult when completing assignments:
  ▪ The Textbook
  ▪ Lecture Notes
  ▪ Resources listed on the course website
  ▪ Python3 documentation
  ▪ Office Hours, TAs, Instructors
Homework 0 Due Monday In-class

• Instructions for your Anonymous ID is on there

• REMEMBER THIS NUMBER
  ▪ (useful for homeworks and exams)
Accessing the Lab

• Labs are in TCL 217A (behind the stairwell, up the ramp)

• This door is also always locked!
  ▪ The combination is 1-2-4-8-1-6 (think 1-2-4-8-16...powers of 2)

• Lab starts on Monday (or Tuesday, depending on your schedule!)
Laptop Policy

Lectures

• There is no official laptop/computer policy for lectures, but...

• Research literature finds that multi-tasking laptop use in the classroom results in less learning / lower test performance [1]

• And that it also results in less learning / test performance for those in direct view of another student’s multi-tasking laptop use [1]


More literature: https://cs.brown.edu/courses/cs019/2018/laptop-policy.html#%28part._rsch%29
Laptop Policy

In other words:

Not only does using a laptop in class for off-task behavior negatively impact your learning...

...it also negatively impacts the learning of every student who can see your screen!
“WITH GREAT POWER COMES GREAT RESPONSIBILITY.”

– Uncle Ben

Please think twice before multi-tasking on your laptop during lecture!

- Iris Howley
Hello world

Writing our first python program.
Installing Python

• Checking version of Python on machine (Mac, Linux)
  • `python --version`

• For this class, we need Python 3.6.4 or above

• Installing Python3 on your machine
  • https://www.python.org/downloads/

• Preinstalled on all CS lab machines

• If your personal machine is Windows
  • It’s possible, but Iris & Shikha won’t be any help getting it to work

• Recommend doing all lab work on machines in the CS labs instead
# (c) 2019 iris howley
# write my first-ish python program

print('Hello, earth!')
Computer Science Colloquium Today!
(and pretty much every Friday)

• 2:35pm in TCL123 (Wege Auditorium)

• Faculty Research Presentations

• + snacks

• Diverse interests and background!
• Short “lightning” talks today
• Should follow-up individually for more info!
QUESTIONS?
Leftover Slides
With a few lines of code, we can also count the number of words in a book, like Tom Sawyer.
# (c) 2019 iris howley
# counts the number of words in tom sawyer

# open the file
book=open("tomsawyer.txt")

# counts num words by splitting on spaces
num_words = len(book.read().split())
print(num_words)

book.close()

# another way to count
book=open("tomsawyer.txt")
count = 0
for x in book.read().split(): count+=1
print(count)

book.close()
What is computer science?

• Computer programming is not computer science!
  ▪ Computer programming is a tool for computer scientists to communicate

• Computer scientists don’t just solve problems for a PC/Mac or in Python, but more abstract problems
  ▪ What is a microwave?
  ▪ It has an on/off button, a door, a plug...a box that cooks things with a door
  ▪ This is abstraction and it’s one of computer scientist’s greatest tools