Lecture #01 – Course Introduction, What is Computation
Lecture Overview

• Course introduction

• What is computation

Disclaimer: Much of the material and slides for this lecture were borrowed from — Ruth Anderson, Michael Ernst and Bill Howe’s CSE 140 class
Course Information
Course Staff

• **Instructors:** Fuat Akal, Aykut Erdem, Erkut Erdem

![Instructors' photos]

• **Teaching Assistants:**
  – Necva Bolucu
  – Bahar Gezici
  – Yunus Can Bilge

Do not hesitate to ask TAs for help!
About BBM 101

• This course teaches core programming concepts with an emphasis on data manipulation tasks from science, engineering, and business

• **Goal** by the end of the semester: Given a data source and a problem description, you can independently write a complete, useful program to solve the problem

• **BBM103 Introduction to Programming Laboratory I**
  – Students will gain skills to apply the concepts to real world problems
Learning Objectives

• Computational problem-solving
  – Writing a program will become your “go-to” solution for data analysis tasks.

• Basic Python proficiency
  – Including experience with relevant libraries for data manipulation, scientific computing, and visualization.
What This Course is not

• A “skills course” in Python
  – ...though you’ll become proficient in the basics of the Python programming language
  – ...and you will gain experience with some important Python libraries

• A “project” course
  – the assignments are “real,” but are intended to teach specific programming concepts

• A “software engineering” course
  – Programming is the starting point of computer science and software engineering
Communication

• Website: http://web.cs.hacettepe.edu.tr/~bbm101/

• See the website for all administrative details

• Read the handouts and required texts, *before* the lecture

• Take notes!

• Always use your department account while sending email!

• Follow the course in Piazza
  https://piazza.com/hacettepe.edu.tr/fall2019/bbm101
Text Books

• **The Python Tutorial**, available from the Python website.
  – This is good for explaining the nuts and bolts of how Python works.


• **Think Python, 2nd edition**
  – Freely available online in [HTML](#) and [PDF](#).
  – Also available for purchase as a printed book, but don't buy the first edition.
  – This book introduces more conceptual material, motivating computational thinking.

• There is an interactive version of “How to Think Like a Computer Scientist” (the first edition of “Think Python”), which lets you type and run Python code directly while reading the book.
Grading Policy

• Grading for BBM 101 will be based on
  – two midterm exams (25+30=55%)
  – final exam (40%)
  – class participation (5%)

• In BBM 103, the grading will be based on
  – five assignments (5+10+3*20=75%)
  – six quizzes (25%) (the lowest 1 quiz grade will be dropped)
Attendance

• Attendance to the lectures is mandatory.

• A student who does not attend the lectures more than 4 weeks will fail BBM101 directly with an F1 grade.

• A student who does not attend more than 1 recitation session or does not submit more than 1 assignment will fail BBM103 directly with an F1 grade.
Academic Integrity

• Honest work is required of a scientist or engineer.

• Collaboration policy on the course web. Read it!
  – Discussion is permitted.
  – Carrying materials from discussion is not permitted.
  – Everything you turn in must be your own work.
    • Cite your sources, explain any unconventional action.
  – You may not view others’ work.
  – If you have a question, ask.

• We trust you completely.

• But we have no sympathy for trust violations – nor should you!
How to Succeed

• No prerequisites

• Non-predictors for success:
  – Past programming experience
  – Enthusiasm for games or computers

• Programming and data analysis are challenging

• Every one of you can succeed
  – There is no such thing as a “born programmer”
  – Work hard
  – Follow directions
  – Be methodical
  – Think before you act
  – Try on your own, then ask for help
  – Start early
Python
Python Version

• Whatever IDE you choose to work with, always stick to **Python version 3.6.5 or higher**

• **Always** use this version to code your assignments.
Integrated Development Environment (IDE)

• There are many!
Our Recommendation: PyCharm
Computer Programming
Computer are now everywhere!
Computer are now everywhere!

Self Driving Cars

Medical Diagnosis and Imaging

Entertainment

Internet of Things

Recommendation Systems

Finance
“It’s a great time to be a data geek.”
-- Roger Barga, Microsoft Research

“The greatest minds of my generation are trying to figure out how to make people click on ads”
-- Jeff Hammerbacher, co-founder, Cloudera
All of Science is Reducing to Computational Data Manipulation

Old model: “Query the world” (Data acquisition coupled to a specific hypothesis)

New model: “Download the world” (Data acquisition supports many hypotheses)

- Astronomy: High-resolution, high-frequency sky surveys (SDSS, LSST, PanSTARRS)
- Biology: lab automation, high-throughput sequencing,
- Oceanography: high-resolution models, cheap sensors, satellites

40TB / 2 nights

~1TB / day
100s of devices
Example: Assessing Treatment Efficacy

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td>fu_2wk</td>
<td>fu_4wk</td>
<td>fu_8wk</td>
<td>fu_12wk</td>
<td>fu_16wk</td>
<td>fu_20wk</td>
<td>fu_24wk</td>
<td>total4type_fu</td>
<td>clinic_zip</td>
<td>pt_zip</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>4</td>
<td>7</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>12</td>
<td>98405</td>
<td>98405</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>98405</td>
<td>98403</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>98405</td>
<td>98445</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>8</td>
<td>98405</td>
<td>98332</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>98405</td>
<td>98405</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>98405</td>
<td>98406</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>14</td>
<td>98405</td>
<td>98418</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>98499</td>
<td>98404</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>98405</td>
<td>98402</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>98405</td>
<td>98405</td>
</tr>
<tr>
<td>11</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>98404</td>
<td>98404</td>
</tr>
<tr>
<td>12</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>98499</td>
<td>98498</td>
</tr>
<tr>
<td>13</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>98499</td>
<td>98498</td>
</tr>
<tr>
<td>14</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>98499</td>
<td>98498</td>
</tr>
<tr>
<td>15</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>98405</td>
<td>98405</td>
</tr>
<tr>
<td>16</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>5</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>98499</td>
<td>98405</td>
</tr>
<tr>
<td>17</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>98499</td>
<td>98498</td>
</tr>
</tbody>
</table>

Number of follow ups within 16 weeks after treatment enrollment.

Question: Does the distance between the patient’s home and clinic influence the number of follow ups, and therefore treatment efficacy?
# This program reads an Excel spreadsheet whose penultimate
# and antepenultimate columns are zip codes.
# It adds a new last column for the distance between those zip
codes, and outputs in CSV (comma-separated values) format.
# Call the program with two numeric values: the first and last
# row to include.
# The output contains the column headers and those rows.

headers = sheet.row_values(0) + ["distance"]
print comma_separated(headers)

for rownum in range(first_row,row_limit):
    row = sheet.row_values(rownum)
    (zip1, zip2) = row[-3:-1]
    if zip1 and zip2:
        # Clean the data
        zip1 = str(int(zip1))
        zip2 = str(int(zip2))
        row[-3:-1] = [zip1, zip2]
        # Compute the distance via Google Maps
        try:
            distance = gd.query(zip1,zip2).distance
        except:
            print >> sys.stderr, "Error computing distance:",
        zip1, zip2
        distance = ""
        # Print the row with the distance
        print comma_separated(row + [distance])
        # Avoid too many Google queries in rapid succession
        time.sleep(random.random()+0.5)
Some statistics (from U.S.)

The value of a computer science education

$0.58M
lifetime earnings of a high school graduate*

$1.19M
lifetime earnings of a college graduate*

$1.67M
lifetime earnings of a computer science major*

*Net present value today

Source: Brookings

Slide credit: code.org
Computing jobs are the #1 source of new wages in the United States

500,000 current openings: These jobs are in every industry and every state, and they’re projected to grow at twice the rate of all other jobs.
Some statistics (from U.S.)

The STEM* problem is in computer science:

- 71% of all new jobs in STEM are in computing
- 8% of STEM graduates are in computer science

Sources: Bureau of Labor Statistics, National Center for Education Statistics

*STEM = Science, Technology, Engineering, and Math
What is meant by computation?
Some may think:

Computer science is just about learning technology
Some may think:

Computer science is just about learning technology

Computer science is about logic, problem solving, and creativity

Slide credit: code.org
First computer: 1943
First computer: 1943
First computer program: 1843
The Map of Computer Science

https://www.youtube.com/watch?v=SzJ46YA_RaA
What is Knowledge?

• **Declarative knowledge**
  – Axioms (definitions)
  – Statements of fact

  “y is the square root of x if and only if y*y = x”

does not help to find the square root!
What is Knowledge? (cont’d.)

• **Declarative knowledge**
  - Axioms (definitions)
  - Statements of fact

  “y is the square root of x if and only if y*y = x” does not help to find the square root!

• **Imperative knowledge**
  - How to do something
  - A sequence of specific instructions (what computation is about)

**Babylonian method**
Get x as an input
1. Begin with an arbitrary positive number \( y_0 \) (an initial guess)
2. If \( y_n^2 \approx x \), stop
   Else let \( y_{n+1} = (y_n + x/y_n)/2 \) (found the solution - \( y_n \)) (use the arithmetic mean to approximate the geometric mean)
3. Repeat step (2)
What is Knowledge? (cont’d.)

• Another example – Estimating greatest common divisor (gcd)

**Declarative definition**
“d is the gcd of a and b if and only if d is the largest possible integer satisfying a = d*x and b = d*y with x and y being two positive integers”

**Imperative definition: Euclid’s formula**
Get 2 positive integers a and b, a>=b as input
1. Divide a by b, call the remainder R
2. If R = 0, stop
   Else let a = b and b = R
3. Repeat step 2

Use Euclid’s formula to compute gcd(48,18).
What is a Computer?

• A device that executes a sequence of computations and instructions.

• Modern computers are electronic and digital.
Programs

• These sequences of instructions and computations is called a **program**.

• We will be designing programs in this course.

• These programs will be based on **algorithms**.
Where did the Term ‘Computer’ Originate?

- The definition from The Oxford Dictionary:

  “Computer (noun). A person who makes calculations, especially with a calculating machine.”

Courtesy of the Library of Congress
Fixed Program Computers

• Developed to solve a specific problem (set).

• Very old roots, old perspectives, ...
  – Abacus
  – Antikythera Mechanism
  – Pascaline
  – Leibniz Wheel
  – Jacquard’s Loom
  – Babbage Difference Engine
  – The Hollerith Electric Tabulating System
  – Atanasoff-Berry Computer (ABC)
  – Turing Bombe
  – etc.
Abacus (500 BC)

- First pocket calculator
- Still used by businessmen in Asia.
Antikythera Mechanism (100 BC)

- First analog computer
- An ancient mechanical computer designed to calculate astronomical positions
Antikythera Mechanism (100 BC)

- First analog computer
- An ancient mechanical computer designed to calculate astronomical positions

The Antikythera mechanism: decoding an ancient Greek mystery

*NATURE*

Vol 454, Issue 7204
31 July 2008

https://www.youtube.com/watch?v=DiQSHiAYt98
Pascaline (1642)

- Blaise Pascal, 1642
- A mechanical calculator for performing two arithmetic operations: addition and subtraction
Leibniz Wheel (1694)

• Gottfried Wilhelm von Leibniz, 1694
• A mechanical calculator for performing all four arithmetic operations: addition, subtraction, multiplication and division
Jacquard’s Loom (1801)

• Developed in 1801 by Joseph-Marie Jacquard.
• The loom was controlled by a loop of punched cards.
• Holes in the punched cards determined how the knitting proceeded, yielding very complex weaves at a much faster rate.
Babbage Difference Engine (1832)

- Charles Babbage, 1832
- A mechanical calculator designed to tabulate polynomial functions (can be used for solving polynomial equations, curve fitting, etc.)
- A working difference engine was built in 1991 to celebrate the 200th anniversary of Babbage's birth (London Science Museum).
- It could hold 8 numbers of 31 decimal digits each and could thus tabulate 7th degree polynomials to that precision.
The Hollerith Electric Tabulating System

• **1880 Census.** Took 1,500 people 7 years to manually process data.

• **Herman Hollerith.** Developed counting and sorting machine to automate.
  – Use punch cards to record data (e.g., gender, age).
  – Machine sorts one column at a time (into one of 12 bins).
  – Typical question: how many women of age 20 to 30?

• **1890 Census.** Finished months early and under budget!

Hollerith tabulating machine and sorter

Punch card (12 holes per column)

Adopted from: Sedgewick and Wayne
Modern Punch Cards

• Punch cards. [1900s to 1950s]
  – Also useful for accounting, inventory, and business processes.
  – Primary medium for data entry, storage, and processing.

• Hollerith's company later merged with 3 others to form Computing Tabulating Recording Corporation (CTRC); the company was renamed in 1924.
Modern Punch Cards

• Punch cards. [1900s to 1950s]
  – Also useful for accounting, inventory, and business processes.
  – Primary medium for data entry, storage, and processing.

• Hollerith's company later merged with 3 others to form Computing Tabulating Recording Corporation (CTRC); the company was renamed in 1924.
Atanasoff-Berry Computer (ABC) (1939)

• John Vincent Atanasoff and Clifford Berry, 1939-1942
• One of the first electronic digital computing devices
• Designed to solve a system of linear equations
Turing Bombe (1941)

- Alan Turing, 1939
- Developed to crack German Enigma codes during WW II.
Alan Turing

• 1912-1954
• Considered the “father” of modern computer science.
• Presented formalisms for the notions of computation and computability in the 1930’s.
• Worked at Bletchley Park in Great Britain during WWII to develop Collossus to help break the German Enigma Code.
• Developed the notion in 1950 of a test for machine intelligence now called the Turing Test.
• The Turing Award, the highest award in computing, is named in honor of Alan Turing.
Stored Program Computers

• Problem solving

• What if input is a machine (description) itself?

• Universal Turing machines
  – An abstract general purpose computer
Universal Turing Machines

• Tape
  – Stores input, output, and intermediate results.
  – One arbitrarily long strip, divided into cells.
  – Finite alphabet of symbols.

• Tape head
  – Points to one cell of tape.
  – Reads a symbol from active cell.
  – Writes a symbol to active cell.
  – Moves one cell at a time.

• Is there a more powerful model of computation? No!

Most important scientific result of 20th century?

Adopted from: Sedgewick and Wayne
Universal Turing Machines

- **Tape**
  - Stores input, output, and intermediate results.
  - One arbitrarily long strip, divided into cells.
  - Finite alphabet of symbols.

- **Tape head**
  - Points to one cell of tape.
  - Reads a symbol from active cell.
  - Writes a symbol to active cell.
  - Moves one cell at a time.

- Is there a more powerful model of computation? No!

Most important scientific result of 20th century?

Adopted from: Sedgewick and Wayne

https://www.youtube.com/watch?v=E3keLeMwfHY
Questions About Computation

• What is a general-purpose computer?
• Are there limits on the power of digital computers?
• Are there limits on the power of machines we can build?

David Hilbert  Kurt Gödel  Alan Turing  Alonzo Church  John von Neumann

Adopted from: Sedgewick and Wayne
Church-Turing Thesis (1936)

Turing machines can compute any function that can be computed by a physically harnessable process of the natural world.

- **Remark.** "Thesis" and not a mathematical theorem because it's a statement about the physical world and not subject to proof.
- Use simulation to prove models equivalent.
  - Android simulator on iPhone.
  - iPhone simulator on Android.
- **Implications.**
  - No need to seek more powerful machines or languages.
  - Enables rigorous study of computation (in this universe).
- **Bottom line.** Turing machine is a simple and universal model of computation.

Adopted from: Sedgewick and Wayne
Church-Turing Thesis: Evidence

- 8 decades without a counterexample.
- Many, many models of computation that turned out to be equivalent.

<table>
<thead>
<tr>
<th>model of computation</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>enhanced Turing machines</td>
<td>multiple heads, multiple tapes, 2D tape, nondeterminism</td>
</tr>
<tr>
<td>untyped lambda calculus</td>
<td>method to define and manipulate functions</td>
</tr>
<tr>
<td>recursive functions</td>
<td>functions dealing with computation on integers</td>
</tr>
<tr>
<td>unrestricted grammars</td>
<td>iterative string replacement rules used by linguists</td>
</tr>
<tr>
<td>extended L-systems</td>
<td>parallel string replacement rules that model plant growth</td>
</tr>
<tr>
<td>programming languages</td>
<td>Java, C, C++, Perl, Python, PHP, Lisp, PostScript, Excel</td>
</tr>
<tr>
<td>random access machines</td>
<td>registers plus main memory, e.g., TOY, Pentium</td>
</tr>
<tr>
<td>cellular automata</td>
<td>cells which change state based on local interactions</td>
</tr>
<tr>
<td>quantum computer</td>
<td>compute using superposition of quantum states</td>
</tr>
<tr>
<td>DNA computer</td>
<td>compute using biological operations on DNA</td>
</tr>
</tbody>
</table>
Babbage’s Analytical Engine (1834, 1836)

• Designed around 1834 to 1836
  – was to be a universal machine capable of any mathematical computation
  – embodies many elements of today’s digital computer
  – a control unit with moveable sprockets on a cylinder that could be modified
  – separated the arithmetic operations (done by the mill) from the storage of numbers (kept in the store)
    • store had 1000 registers of 50 digits each
  – Babbage incorporated using punched cards for input
    • idea came from Jacquard loom

• Never built by Babbage due to lack of funds and his eventual death in 1871

Adopted from: T. Cortina
Ada Lovelace

- 1815-1852

- Daughter of poet Lord Byron

- Translated Luigi Menabrea’s article on Babbage’s Analytical Engine to English
  - Quadrupled its length by adding lengthy notes and detailed mathematical explanations

- Referred to as the world’s first programmer
  - Described how the machine might be configured (programmed) to solve a variety of problems.
The Zuse Z3 Computer (1941)

• Konrad Zuse, 1941
• The original Z3 was destroyed in a bombing raid of Berlin in 1943.
• Zuse later supervised a reconstruction of the Z3 in the 1960s (currently on display at the Deutsches Museum in Munich)
Colossus Mark 1 (UK,1944)

- The world's first electronic digital computer with programmability.
ENIAC (Mauchly and Eckert, USA, 1946)

- The first large-scale general-purpose electronic computer without any mechanical parts.
- Designed to calculate artillery firing tables for the United States Army's Ballistic Research Laboratory
EDVAC (von Neuman, USA, 1951)

- Unlike the ENIAC, it uses binary rather than decimal numbering system.
- Instructions were stored in memory sequentially with their data.
- Instructions were executed sequentially except where a conditional instruction would cause a jump to an instruction someplace other than the next instruction.
The Computer Tree

Summary

• What is computation?
  – What is knowledge?
  – What is a computer?
  – What is a program?
  – History of computing
The Birth of the Computer

• A TED talk given by George Dyson

http://www.ted.com/talks/george_dyson_at_the_birth_of_the_computer.html