Course Introduction

BBM 101 - Introduction to Programming I

Hacettepe University
Fall 2016

Fuat Akal, Aykut Erdem, Erkut Erdem

1

Welcome to 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

2

Course Staff

- Lecturers:
  - Asst. Prof. Dr. Fuat Akal
  - Asst. Prof. Dr. Aykut Erdem
  - Asst. Prof. Dr. Erkut Erdem

3

Course Staff

- TAs (Teaching Assistants):
  - Necva Bölücü
  - Selma Dilek
  - Burcu Yalçıner
  - Selim Yılmaz

Do not hesitate to ask TAs for help!

4

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.

- Experience working with real datasets
  - astronomy, biology, linguistics, oceanography, open government, social networks, and more.
  - You will see that these are easy to process with a program, and that doing so yields insight.

5

What This Course is not

- A "skills course" in Python
  - …though you will become proficient in the basics of the Python programming language
  - …and you will gain experience with some important Python libraries

- A data analysis / "data science" / data visualization course
  - There will be very little statistics knowledge assumed or taught

- 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

6

"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
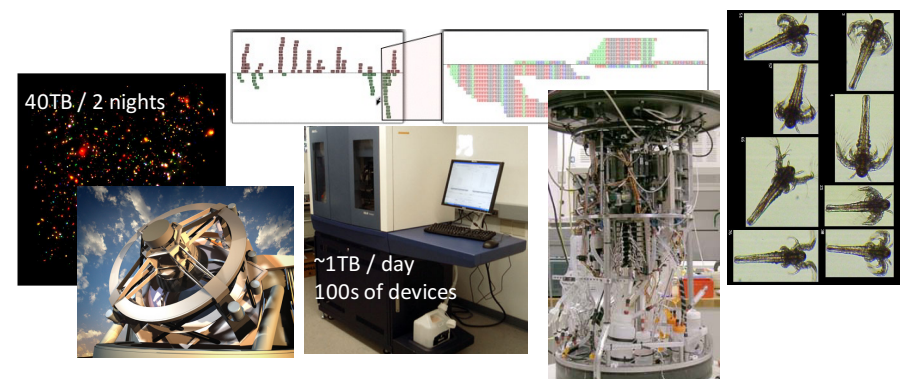
8

Example: Assessing Treatment Efficacy

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

number of follow ups within 16 weeks after treatment enrollment.

Zip code of clinic

Zip code of patient

Question: Does the distance between the patient's home and clinic influence the number of follow ups, and therefore treatment efficacy?

Python Program to Assess Treatment Efficacy

```python
# 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.

# Libraries to use
import random
import sys
import xlrd        # library for working with Excel spreadsheets
import time
from gdapi import GoogleDirections

# No key needed if few queries
gd = GoogleDirections('dummy-Google-key')

wb = xlrd.open_workbook('mhip_zip_eScience_121611a.xls')
sheet = wb.sheet_by_index(0)

# User input:  first row to process, first row not to process
first_row = max(int(sys.argv[1]), 2)
row_limit = min(int(sys.argv[2])+1, sheet.nrows)

def comma_separated(lst):
  return ",".join([str(s) for s in lst])

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)
```
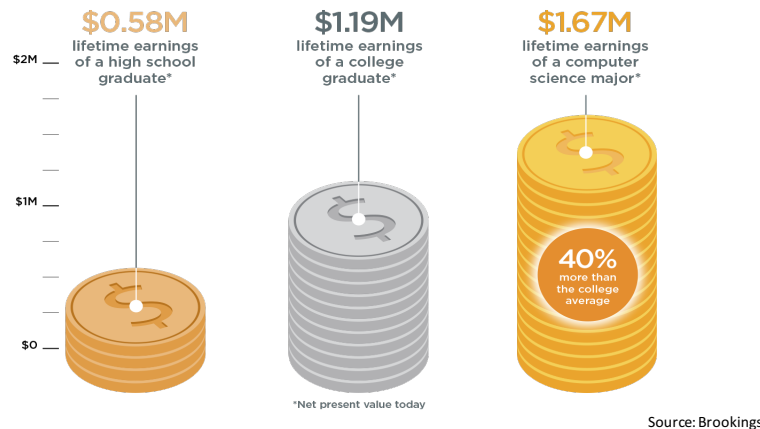
23 lines of executable code!

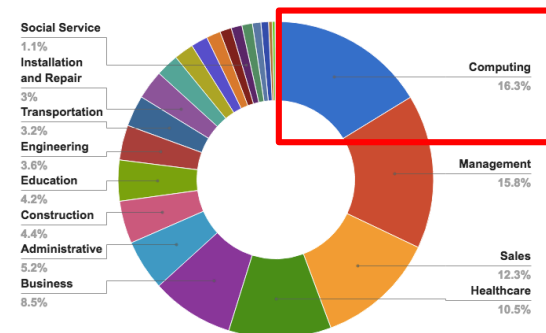
Some statistics (from U.S.)

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*

40% more than the college average

*Net present value today

Source: Brookings

Slide credit: code.org

Some statistics (from U.S.)

### 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.

Slide credit: code.org

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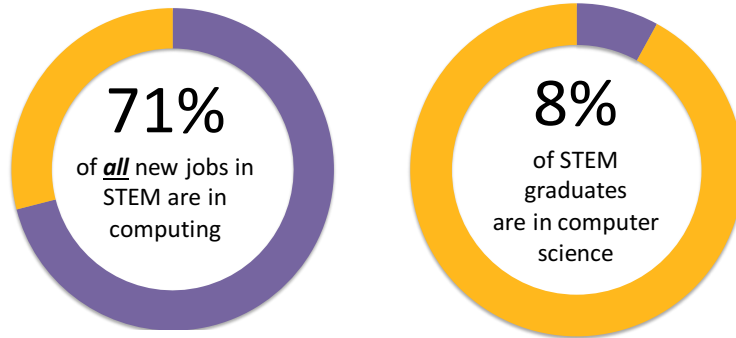
Sources: Bureau of Labor Statistics, National Center for Education Statistics

Slide credit: code.org

*STEM = Science, Technology, Engineering, and Math

Course Logistics

- 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!

- Follow the course in Piazza
  `https://piazza.com/hacettepe.edu.tr/fall2016/bbm101`

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
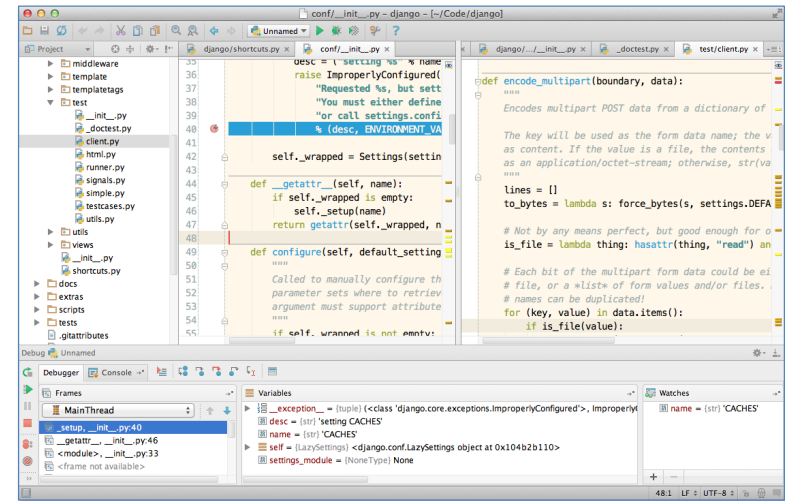 - Try on your own, then ask for help
  - Start early

Integrated Development Environment (IDE)

- There are many!
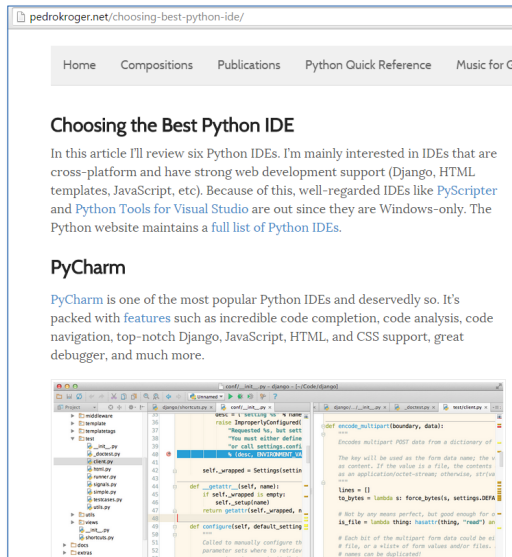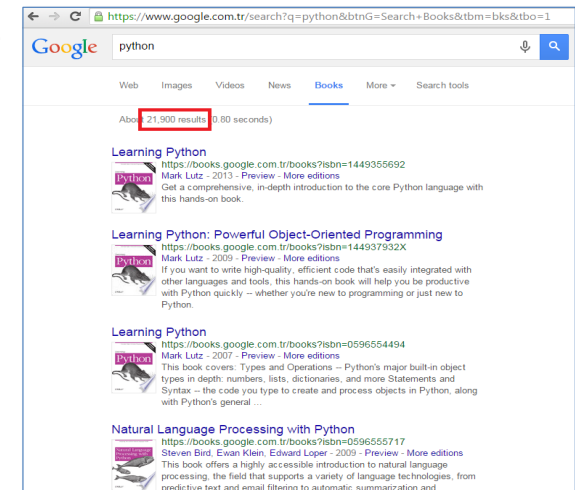

Our Recommendation: PyCharm

Python Version

- Whatever IDE you choose to work with, always stick to **Python version 3.5.2**

- **Always** use this version to code your assignments.

Books

- There are many!

Our Recommendation for Books

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

- Introduction to Computation and Programming Using Python, Second Edition, John V. Guttag, MIT Press, August 2016

- 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.