Compsci 101
Python Code, Variables

Susan Rodger
September 1, 2022

```python
st = f.read().decode('utf-8')
st = st.lower()
total = len(st)
```
B is for ...

• Bug
  • What you will always have and need to fix

• Bits
  • Zeros and Ones, like C,G,A,T makes up DNA

• Byte
  • 8 bits that represent a character

• Boolean
  • Type that's true or false
B is for ...

- **Bug**
  - What you will always have and need to fix
- **Bits**
  - Zeros and Ones, like C,G,A,T makes up DNA
- **Byte**
  - 8 bits that represent a character
    - 01000001 code for letter "A"
- **Boolean**
  - Type that's true or false
Grace Hopper

- Computer Scientist
- Rear Admiral in US Navy
- One of first programmers for one of first computers: Harvard Mark 1
- Handed out nanoseconds
- First computer bug in 1947

"The only phrase I've ever disliked is, 'Why, we've always done it that way. ' I always tell young people, 'Go ahead and do it."

Announcements

• Survey coming out– complete this week
• Lab 0 is Friday, no Prelab
• Prelab 1 before lab– Install Python/Pycharm
  • Ways to get help:
    • Office hours, consulting hours
    • Post on Ed Discussion – what type of machine, etc
    • Install Fest at Co-lab, Sept 6-7, 2pm-8pm
• Ed Discussion Back channel during lecture
• QZ03 and reading due Tuesday at 10:15am
• Assignment 0 - Blockly due 2/8
Is this the right course for you?

• **CompSci 101**
  • beginner
  • little or no programming experience

• **CompSci 201**
  • 4/5 on AP CS A
  • OR Programming Experience in Python or Java or ?
    • Problem solving with arrays or lists
    • Looping structures (while/for)
    • Writing functions/methods
    • Problem solving with Sets, Dictionaries or maps?
Can’t take CompSci 101 if

• You already took CompSci 201, or CompSci 116, or ENG 103 …...

• You won’t get credit for this course

• This is a beginner course
Where to sit? Laptops?

• Sit anywhere but the top 2 seater row and the top 5 full rows.
  • Come forward meet someone

• Laptop policy
  • Use your laptop in class only for CompSci 101
    • No watching sports videos, etc
      – RUDE and distracting to other students
      – Don't come to class if you feel you have to do this
  • Not be doing other coursework
Practice, Practice, Practice
Practice, Practice, Practice
Practice results in Success
Practice results in Success
Don’t get behind!!!

• Difficult to catch up...
Don’t get behind!!!

• Difficult to catch up...
Plan for the Day (PFTD)

• Look at a sample Python Program
  • OK if you don’t understand it all
• How to run Python Code
  • Run complete program in Pycharm
  • Short code segments with Python Console
    • Python Console is in Pycharm
• Names, types, and values in Python
• Functions in Python
Finish Slides from Last Time
Duke Connection: Fred Brooks '53

• What Would FB Say?
"The most important single decision I ever made was to change the IBM 360 series from a 6-bit byte to an 8-bit byte, thereby enabling the use of lowercase letters. That change propagated everywhere."

"Fred Brooks" by Copyright owned by SD&M (www.sdm.de) - Request for picture sent by email to Fred Brooks by uploader (Mark Pellegrini; user:Raul654) Fred sent this photo back, along with contact information for Carola Lauber at SD&M, who gave copyright permission. Licensed under CC BY-SA 3.0 via Wikimedia Commons - https://commons.wikimedia.org/wiki/File:Fred_Brooks.jpg#/media/File:Fred_Brooks.jpg
Why is programming fun?

Fred Brooks

• First is the sheer joy of making things
• Second is the pleasure of making things that are useful
• Third is the fascination of fashioning complex puzzle-like objects of interlocking moving parts
• Fourth is the joy of always learning
• Finally, there is the delight of working in such a tractable medium. The programmer, like the poet, works only slightly removed from pure thought-stuff.
Understanding Code

• We will look at an interesting Python program
  • Try to figure out what it does

• You Likely Will NOT understand all this code
• Maybe none of it

• That’s OK
How Breakout Groups Work with Google form links

• Given a bitly link
  • Type it in OR click on it on the calendar page

• What you should do:
  • Introduce yourselves
  • Each person fills out the google form
  • Includes your email, name and netid
  • Discuss each question and fill out
  • Be mindful of time
```python
import urllib.request

def processURL(url):
    f = urllib.request.urlopen(url)
    st = f.read().decode('utf-8')
    st = st.lower()
    total = len(st)
    print("total # chars = ", total)
    print("total # z's = ", st.count("z"))
    for ch in "abcdefghijklmnopqrstuvwxyz":
        print(ch, st.count(ch))

if __name__ == '__main__':
    processURL("https://www2.cs.duke.edu/csed/data/kjv10.txt")
```
```
import urllib.request

def processURL(url):
    f = urllib.request.urlopen(url)
    st = f.read().decode('utf-8')
    st = st.lower()
    total = len(st)
    print("total # chars = ", total)
    print("total # z's = ", st.count("z"))
    for ch in "abcdefghijklmnopqrstuvwxyz":
        print(ch, st.count(ch))

if __name__ == '__main__':
    processURL("https://www2.cs.duke.edu/csed/data/kjv10.txt")
```

Output:
```
total # chars = 4345018
total # z's = 2977
a 275338
b 48761
c 54774
d 157899
e 411615
f 83377
g 55089
h 282472
i 193510
...```
WOTO-2 Understanding Code
Names, Types, and Values

• Relate to a file. Consider: homework.pdf
• What is its name?
  • What is its type?
• What is its value?
Names, Types, and Values

• Relate to a file. Consider:  homework.pdf
• What is its name?
  • homework.pdf
• What is its type?
  • .pdf (portable document format)
  • File format created by Adobe
• What is its value?
  • Content of the file, your homework for a class?
Names, Types, and Values

• Relate to a file. Consider: cats.jpg
• What is its name?
  • What is its type?
• What is its value?
Names, Types, and Values

• Relate to a file. Consider: cats.jpg
• What is its name?
  • cats.jpg
• What is its type?
  • .jpg (type of image file)
• What is its value?
  • Content of the file, picture of cats?
Numeric Python Building Blocks

• Numbers are not everything! But good start
  • Values and arithmetic expressions
  • Integer aka int: 0, 3, -2, 5, ...
  • Float: 2.5, 3.6673, 1.938e+120
  • Operators: +, -, *, /, **
  • Operators: // and %

• Demo in Python Console
Interactive Console

- Short way to look at Python values and expressions
- Look in the bottom left corner of PyCharm
- Click on “Python Console”
Summary of Numbers

- Integers are arbitrarily large in Python 3
- Float values do not have infinite precision
  - Floats are for decimal values

- Be attentive to parentheses and precedence
- Understand / and // and %
  - Modulus or remainder
Python Strings

• A string is a sequence of characters
  • String literals use single or double quotes
  • "hello" and 'world' are both strings

• Operators we'll use: + and [:]
  • Concatenation and Slicing
  • Adding and taking apart?
    • *Today just adding*

• Demo in Python Console
Types and Conversion

• How do you convert a .jpg to a .png?

• Can we add a string and an integer?
Types and Conversion

• **How do you convert a .jpg to a .png?**
  • Change the bits from one format to another

• **Can we add a string and an integer?**
  • What does 5 + "cow" mean?
  • What does 5 * "cow" mean?
  • Why?

• **Python Console Demo**
Using Python Console

• Not writing a whole program
• Just checking out values or writing simple code

• What is the difference in Python Console of:
  >>> print("a" + " " + "b")

  >>> "a" + " " + "b"
Using Python Console

• Not writing a whole program
• Just checking out values or writing simple code

• What is the difference in Python Console of:

```python
>>> print(“a” +” “ + “b”)
a b

>>> “a” + “ “ + “b”
’a b’
```
Variables

• We use variables to store values so we can use them and re-use them in expressions
  • Name associated with storage (spot in memory)
  • Assign value to a variable

• How to read: num = 5, word = "hello"
  • Why say 'gets' or 'is assigned' and not 'equals’
  • We’ll use ‘equals’ later to mean equality
Variable idea

1) num = 6
Variable idea

1) \( num = 6 \)
Variable idea

1) num = 6
Variable idea

2) $y = \text{num} + 4$

Computer

num

6
Variable idea

2) \[ y = \text{num} + 4 \]
Variable idea

2) \( y = \text{num} + 4 \)
Variable idea
3) \( num = y \times 2 \)
Variable idea

3) \( num = y \times 2 \)

Computer

\( y \) | 10
\( num \) | 6
Variable idea

3) \[\text{num} = \text{y} \times 2\]
Anatomy of a variable

• **Variables in Python have a type, changeable**
  • Initially `var = 5`, change to `var = “hello”`
  • Use the `type(...)` function to determine type, but documentation/comments are better

• **Variables are names/labels, references to an object stored elsewhere (basically)**
  • My address is “202 Main Street”
  • That’s the name/label, my house is elsewhere
  • For `var = “hello”`, the string is elsewhere
Subtleties

- Variables on LHS and RHS
  - Value compared to Name
  - LHS – Left Hand Side
  - RHS – Right Hand Side

- What happens here?
  - Value compared to Name

- In expressions? What is value

\[
\begin{align*}
\text{num1} &= 17 \\
\text{num2} &= \text{num1} + 12 \\
\text{var1} &= 17 \\
\text{var2} &= \text{var1} + 12 \\
\text{var1} &= "hi" \\
\text{var2} &= \text{var1} \times 3
\end{align*}
\]
Subtleties

- Variables on LHS and RHS
  - Value compared to Name
  - LHS – Left Hand Side
  - RHS – Right Hand Side
  - 1) Evaluate RHS
  - 2) Store in LHS

- What happens here?
  - Value compared to Name

- In expressions? What is value

```
num1 = 17
num2 = num1 + 12
num1 gets 17
num2 gets 29

var1 = 17
var2 = var1 + 12
var1 = "hi"
var2 = var1 * 3
var1 gets 17
var2 gets 29
var1 gets "hi"
var2 gets "hihihi"
```
Basic Python