CompSci 101
Introduction to Computer Science

February 3, 2015
Prof. Rodger

Lecture given by Elizabeth Dowd
Announcements

- Reading RQ08 for next Assignment
- Assignment 3 due today, Asg 4 out
- APT 3 is due on Thursday
- Exam 1 – Feb 12
  - try practice test by Feb 10
  - Accommodations? Reschedule? Fill out forms!
- Prof Rodger – at ACM India conference

- Finish lecture notes from last time
Assignment 4 – Transform

At the far end of town
Where the Grickle grass grows

At-way e-thay ar-fay end-way of-way own-tay
ere-Whay e-thay ickle-Gray ass-gray ows-gray
Lists

• A list is a collection of objects
  
  ```
  scores = [99, 78, 91, 84]
  allAboutMe = [“Mo”, 25, “934-1234”]
  ```

• Lists are *mutable* – use `[num]` to change a value
• Lists are indexed starting at 0, or -1 from the end
• Functions: max, min, len, sum
• Slice lists [::]
List Examples

scores = [10, 8, 10, 9]
print scores
scores[2] = 5
print scores
print max(scores)
print len(scores)
print sum(scores)
print scores[1:]
print scores[1]
List before/after modification

score = [10, 8, 10, 9]

score [2] = 5
Processing List Items

• Process all the items in a list, one item at a time
• Format: \texttt{for variable in list:}
  \hspace{1cm} \texttt{block}

• Example:

\begin{verbatim}
sum = 0
nums = [6, 7, 3, 1, 2]
for value in nums:
    sum = sum + value
print sum
\end{verbatim}
names = ['jo', 'mo', 'bo']
club = names
team = names[:]
names[1] = 'flo'
print names
print club
print team
Making Decisions

Question

True

if block

False
Making Decisions in Python

if condition1:
    Block of code to do if condition is true

elif condition2:
    Block of code to do if condition1 false, condition2 is true

else:
    Block of code to do if other conditions false

• Can have many elifs, leave out elif, leave out else
Making Decisions tools

- Boolean values: True, False
- Boolean operators: and, or, not

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>X and Y</th>
<th>X or Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
</tr>
<tr>
<td>True</td>
<td>False</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>False</td>
<td>True</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
</tr>
</tbody>
</table>

- Relational operators: <, <=, >, >=
- Equality operators: ==, !=
- Look at if examples: miscIf.py
best = "UNC Blue Devils"
print best
if best[:3] == "UNC":
    best = "Duke" + best[3:]
print best

print "num 1 test"
num = int(raw_input("Enter Num: "))
if num > 15:
    print "biggest"
elif num > 10:
    print "bigger"
elif num < 5:
    print "smaller"
else:
    print "middle"

print "num 2 test"
num = int(raw_input("Enter Num: "))
if num > 15:
    print "biggest"
if num > 10:
    print "bigger"
elif num < 5:
    print "smaller"
else:
    print "middle"
More on lists

• range (1,11)
  – Generates a list of numbers from 1 to 10

• Example:
  
  ```python
  for num in range(1, 11):
    print(num)
  ```
Example

answer = 0
for num in range(1,10):
    if num % 3 == 0:
        answer = answer + num
    else:
        answer = answer + 1
Dissect the for loop

```python
for VARIABLE in STRUCTURE:
    BODY
```

Repeat the BODY with the VARIABLE equal to each item in structure
What can the structure be? Variable be?

- STRUCTURE → Variable
- String → character
- List → item in list

- There are other types of structures we will see
Examples

```python
answer = 0
for w in range(5,0,-1):
    answer = answer * 10 + w

answer = ""
word = "NCStateFair"
for some in word:
    answer = answer + " " + some
```
Reading from Files

- Must open file, close file
- Easiest way, read one line as a string and then process it

```python
inputfile = open("datafile.tex")
for line in inputfile:
    line = line.strip()
    """ do something with line ""

inputfile.close()
```
Dissect the for loop (again)

for VARIABLE in STRUCTURE:
    BODY

inputFile = open(“somefile.txt”)
for str in inputFile:
    process str
Writing to File

• Must open file, close file
• Open file for writing

```python
outfile = open("outputfile.txt", 'w')
phrases = ["hello there", ...
for phr in phrases:
    outfile.write(phr + "\n")

outfile.close()
```

Note: refresh to see the file
Exercise with files

- Snarf code
- Uppity.py