ENGL 516X:
Methods of Formal Linguistic Analysis

Semester: Spring '18

Instructor: Sowmya Vajjala

Iowa State University, USA

23 Jan 2018

Class outline

- Week 2 Recap
- Assignment 1 Discussion
- Assignment 2 Description
- Logical Expressions
- Conditional Statements
- Writing programs with conditional statements

Ref: Chapter 3 in the textbook

Recap

Recap: type of a variable

What is the type of the following variables?

1. var1 = "python"
2. var2 = 3.2
3. var3 = 9
4. var4 = 9.0
5. var5 = '3.2'
6. var6 = True
7. var7 = 'b3'
8. var8 = true

Recap: type of a variable

What is the type of the following variables?

1. var1 = "python"
2. var2 = 3.2
3. var3 = 9
4. var4 = 9.0
5. var5 = '3.2'
6. var6 = True
7. var7 = 'b3'
8. var8 = true

What built-in function is useful to know the type of a varialble?

Recap: input()

Let us take this line:
i = input("Enter a number: ")
How do you make sure this variable i is a number and not a string?

Recap: Operator Precedence

Evaluate the following expressions:

1. (3*5)+2-3+(2*4)

Recap: Operator Precedence

Evaluate the following expressions:

1. (3*5)+2-3+(2*4)
   The answer is: 22
2. 1+3**2*4-6

Recap: Operator Precedence

Evaluate the following expressions:

1. (3*5)+2-3+(2*4)
   The answer is: 22
2. 1+3**2*4-6
   The answer is 31.
3. 2*5/6*4+1-3
   The answer is 4.66666....

Recap: Programs from last class - 1

- Description: Program prompts for username, and country, and prints a message.
- One solution posted on discussion forum:

```
name=input("What is your name?")
country=input("What country are you from?") + "."
print("Hello ", name, "from", country, "Welcome to Ames.")
phone=input("Please enter your 10-digit phone number with no spaces or oth
print(phone)
```

Questions raised in the forum

- How to make sure it is a valid phone number? - you will need more experience before being able to do that. May be 3-4 more weeks?
- Tim used len(x) to check if it is 10 characters long.

Recap: Programs from last class - 2

- Description: Compound interest calculation
- One solution posted on discussion forum:

```
P=12000
n=12
r=.08
t=input("What is the number of years?")
t=int(t)
A=P*((1+(r/n))**(n*t))
print(A)
```

Assignment 1 Discussion

Assignment 1 - Key (1)

- print("I am a graduate student in ISU. \n I am learning to program in Python")

Assignment 1 - Key (1)

- print("I am a graduate student in ISU. \n I am learning to program in Python")
- String1+String2 gives "IowaAmes"; String1*Num2 prints string1 num2 times. Remaining two cases fail.

Assignment 1 - Key (1)

- print("I am a graduate student in ISU. \n I am learning to program in Python")
- String1+String2 gives "IowaAmes"; String1*Num2 prints string1 num2 times. Remaining two cases fail.
- max(), min() works for numbers and strings, but not when we are comparing number with a string.

Assignment 1 - Key (1)

- print("I am a graduate student in ISU. \n I am learning to program in Python")
- String1+String2 gives "IowaAmes"; String1*Num2 prints string1 num2 times. Remaining two cases fail.
- max(), min() works for numbers and strings, but not when we are comparing number with a string.
- max("Language","Linguistics"); min("Language","language") - What exactly is happening?

See the document in Canvas. Don't start on the friday before submission.

Assignment 1 - Key (2)

Let us say I am developing a small web-application which collects datafrom students. I ask students two questions - a) Enter a number, b)Enter another number. Once the students enter two numbers, I willshow them the sum of these two numbers. Now, you all know howstudents are. They don't always read instructions properly, and evenif they read, there are those mischevious people. Write an algorithmor draw a flowchart to explain how do you make sure you ensure theyentered only numbers, not names or something.

Assignment 1 - Key (3)

Let us say we live in a world where all plurals in English are formedby adding an -es at the end of a word. Write an algorithm or drawa flow-chart for a possible program that takes a word as input and shows its plural as output.

Assignment 2 Description

Assignment 2

- Grade: 10%
- Deadline: 3 Feb
- Num. Questions: 4 (2+2+2+4)
- About?: What you learn until Today.

See the document in Canvas. Don't start on the friday before submission.

Logical Operators and Boolean Expressions

Boolean Expressions

- What is a Boolean Expression?

Boolean Expressions

- What is a Boolean Expression?
  Something that is either true or false, and where there is no other possibility.
- Where are boolean expressions useful?

Boolean Expressions

- What is a Boolean Expression?
  Something that is either true or false, and where there is no other possibility.
- Where are boolean expressions useful?
  When you want to compare two things and make a decision based on the true/false situation.
- Boolean expressions in Python:
  1. x == y, x!= y (to check if x and y have the same value)
  2. x > y, x >= y
  3. x < y, x<=y
  4. x is y (x is the same as y)
  5. x is not y (x is not the same as y)

Quiz Boolean Expressions

1. Question: What is the difference between $=$ and $==$ symbols?

Quiz Boolean Expressions

1. Question: What is the difference between = and == symbols?
   = is assignment. == is comparison.
2. Question: What is the difference between == and **is**?

Quiz Boolean Expressions

1. Question: What is the difference between = and == symbols?
   = is assignment. == is comparison.
2. Question: What is the difference between == and **is**?
   == is comparison, as usual. "is" is a comparison that compares not the values, but the memory locations of these objects/variables.

Tip: Avoid using **is** for comparing strings and numbers.

Quiz Boolean Expressions

What will these operations below give you: True or False?

1. 5 >= 4.9999999
2. 5 <= "5"
3. a = "string"
   b = "string"
   a==b
4. x = 3
   x==3
5. "tail" < "trail"
6. "tango" < "mango"
7. 2 > 5

Logical Operators

Logical Operators

Three logical operators you should know: and, or, not

Using logical operators between two expressions results in a boolean value.

1. AND : (expression X) and (expression Y) is true only if both the expressions are true.
   E.g., if x=3, which of the following are true:
   - x $>$ 2 and x<=6
   - x $>$ 0 and x <=2

Logical Operators

Three logical operators you should know: and, or, not
Using logical operators between two expressions results in a boolean value.

1. AND : (expression X) and (expression Y) is true only if both the expressions are true.
   E.g., if x=3, which of the following are true:
   - x > 2 and x<=6
   - x > 0 and x <=2
2. OR: expression X) or (expression Y) is true if any of the expressions is true.
   E.g., if x=3, which of the following are true:
   - x > 2 or x<=6
   - x > 0 or x <=2
   - x ==4 or x >56

Logical Operators

Three logical operators you should know: and, or, not
Using logical operators between two expressions results in a boolean value.

1. AND : (expression X) and (expression Y) is true only if both the expressions are true.
   E.g., if x=3, which of the following are true:
   - x > 2 and x<=6
   - x > 0 and x <=2
2. OR: expression X) or (expression Y) is true if any of the expressions is true.
   E.g., if x=3, which of the following are true:
   - x > 2 or x<=6
   - x > 0 or x <=2
   - x ==4 or x >56
3. NOT: not(expression X) is true if expression X is false.
   E.g., if x=3, which of the following are true:
   - not(x>2)
   - not(x<3)

Conditional Statements in Python

Conditional Statements

The Rhombuses of flowcharts

- They are the if(A), do B. Else, do C kind of statements.
- They allow us to model decisions inside our program.
- They can be as simple as a single if-else or a chain of conditions or conditions within conditions and so on.
- Very useful construct in writing your programs

Conditional Statements in Python

1. Simple conditional statement:

```
if x==1:
   print("x is one")
else:
   print("x is anything except one")
```

2. Chained conditional statement (testing for several conditions in sequence):

```
if x==0:
   print("x is zero")
elif x==1:
   print("x is one")
else:
   print("x is neither zero nor one")
```

Note: Look at the indentation of the code.

Nested Conditionals

Conditions within conditions.

```
if x == y:
    print 'x and y are equal'
else:
    if x < y:
        print 'x is less than y'
    else:
        print 'x is greater than y'
```

A Small Recap exercise

Write the flowchart equivalents of:

1. Simple conditional
2. Chained simple conditionals
3. Nested conditionals

Combining Conditions and Logical Operators

Examples

- if x>2 and x<6:
- if (x>10 and x <20) or y=="operation":
- if y=="operation" and z==False:
- if not(y=="operation") and (x==True or z<25):

.. and so on.

ESL Flowchart problem as a program - 1

Description from last week:

- Draw a flowchart for a ESL class placement test.
- Here are thedetails:Everyone comes and takes an online exam with 40 fill in theblank questions that test English grammar skills.
- Based on the test, you have to put the test takers into one of the sections: Beginner, Intermediate, Upper intermediate, advanced.
- Let us say anyone who scores less than 10 is a beginner, between 11 and 20 is in intermediate state, 21-30 is Upperintermediate, higher than 30 is advanced.

ESL Placement: Program

```
score = input("What is the student's score: ")
if score.isnumeric():
  score = int(score)
  if score >=0 and score <=40:
    if score <=10:
       print("Beginner")
    elif score >=11 and score <=20:
       print("Intermediate")
    elif score >=21 and score <=30:
       print("Upper Intermediate")
    else:
       print("Advanced")
  else:
    print("Score is not in the range 0-40")
else:
    print("You did not enter a number!")
```

ESL Placement: Program

```
score = input("What is the student's score: ")
if score.isnumeric():
  score = int(score)
  if score >=0 and score <=40:
    if score <=10:
       print("Beginner")
    elif score >=11 and score <=20:
       print("Intermediate")
    elif score >=21 and score <=30:
       print("Upper Intermediate")
    else:
       print("Advanced")
  else:
    print("Score is not in the range 0-40")
else:
    print("You did not enter a number!")
```

Question: Why did I not write another if score $>=$ 31 and score $<=$ 40?

ESL Placement: Program

```
score = input("What is the student's score: ")
if score.isnumeric():
  score = int(score)
  if score >=0 and score <=40:
    if score <=10:
       print("Beginner")
    elif score >=11 and score <=20:
       print("Intermediate")
    elif score >=21 and score <=30:
       print("Upper Intermediate")
    else:
       print("Advanced")
  else:
    print("Score is not in the range 0-40")
else:
    print("You did not enter a number!")
```

Question: Why did I not write another if score $>=$ 31 and score $<=$ 40? Any questions from your end?

Today's Exercise

1. Write a program that asks the user to enter a temparature in Celsius, and prints the temparature in Fahrenheit (Textbook question).
2. Write a program that asks the user to enter a temparature in Fahrenheit and prints the temparature in Celsius.

Hint 1: Celsius to Fahrenheit conversion: (user_input*9/5)+32 = your answer.
Hint 2: Fahrenheit to Celsius conversion: (user_input-32)*5/9 = your answer.

Next Class

- Topics: Writing our own functions in Python, Practice.
- Readings: Chapter 4 in the text book.