CS 160
CS Orientation

Expressions, Data Types, & Input/Output
Assignment 3: Semantics vs. Syntax

• What are these words?
• Why is this important?
Syntax
Syntax and Semantics...
Which is more important?
What does this have to do with CS?

• How is this like CS programs?
Data Types/Converting

- string - “string of text”
- character – ‘a’
- integer – 79
- boolean – True
- float – 79.0
boolean=True #booleans are always 0 for false or 1 for true
ch='a'  #only strings in python with single or double quotes

print(num+boolean) #can add number and boolean
print(num+ord(ch)) #can add number and ascii value of character
print(chr(49))  #can find out character based on ascii value
print(bool(num)) #any number other than 0 is true
print(int(True)) #True is always 1
Python Operators and Their Computer Symbols

- +
- -
- *
- /
- // — integer division
- %
- **

Python 2

5 / 2
operands
operator
Hierarchy of Operations

- Functions, i.e. `math.sqrt()`
- Power
- Mod
- Mult, Div
- Add, Sub
- Relational
- Logical
Python Examples

- $5 \times 2 + 3 - 10$
- $5 \times (2 + 3) - 10$
- $5/2 \quad \rightarrow \quad 2.5$
- $5//2 \quad \rightarrow \quad 2$

OSU Oregon State University
How about storing values?

• ch = ‘a’
• num = 79
• cont = True
• num = num + 1
• Can we mix types?
  – num = num + cont
  – num = num + ch
  – ch = ch + num
## Expressions and Equations

<table>
<thead>
<tr>
<th>Expressions</th>
<th>Equations</th>
</tr>
</thead>
<tbody>
<tr>
<td>( A + B )</td>
<td>( C = A + B )</td>
</tr>
<tr>
<td>( A ) and ( B ) are numeric. The resultant is numeric and is not stored.</td>
<td>( C ), ( A ), and ( B ) are numeric. The resultant is stored in ( C ).</td>
</tr>
<tr>
<td>( A &lt; B )</td>
<td>( C = A &lt; B )</td>
</tr>
<tr>
<td>( A ) and ( B ) are numeric, character, or string. The resultant is logical and is not stored.</td>
<td>( A ) and ( B ) are numeric, character, or string. The resultant is stored in ( C ); ( C ) is logical.</td>
</tr>
<tr>
<td>( A ) OR ( B )</td>
<td>( C = A ) OR ( B )</td>
</tr>
<tr>
<td>( A ) and ( B ) are logical. The resultant is logical and is not stored.</td>
<td>( C ), ( A ), and ( B ) are logical. The resultant is stored in ( C ).</td>
</tr>
</tbody>
</table>
Sequential Logic Structure

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Flowchart</th>
<th>Pseudocode</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>&lt;Diagram&gt;</td>
<td></td>
</tr>
</tbody>
</table>

5. Instruction
6. Instruction
7. Instruction
8. Instruction

Instruction

Instruction

Instruction
Python Sequential Logic

print(" * ")
print(" *** ")
print(" ***** ")
print(" ******* ")
print("*********")
boolean=True  #booleans are always 0 for false or 1 for true
ch='a'    #only strings in python with single or double quotes

print(num+boolean)  #can add number and boolean
print(num+ord(ch))  #can add number and ascii value of character
print(chr(49))     #can find out character based on ascii value
print(bool(num))   #any number other than 0 is true
print(int(True))

#can't use a variable without assigning a value/defining first
#sequence and order matters!!!
print(num+num2)
num2 = 30
Decision Logic Structure

5. If <decision>
   then
   Instruction
   else
   Instruction
6. ...

If <decision>
   then
   Instruction
else
   Instruction
Endif
   ...

Algorithm | Flowchart | Pseudocode
-----------|-----------|-----------
:::          |          |            

Instruction  | Instruction | Instruction
-----------|------------|-------------

F  | Decision Instruction | T
Relational Operators and Symbols

- >
- >=
- <
- <=
- ==
- !=

*all are binary*  
*2 operands*